Universidad Carlos III de Madrid

Escuela Politécnica Superior

Trabajo de Fin de Grado
Grado en Ingeniería en Informática

Análisis, diseño e implementación del clasificador de opiniones TBONTB

Septiembre 2014

Autor: Manuel José Lazo Reyes
Tutor: Alejandro Calderón Mateos

“Los malos artistas copian. Los buenos roban.”

- Pablo Picasso

Agradecimientos

Quizás de todo lo que he escrito en este documento, esto es lo más difícil porque no soy capaz de expresar todo el agradecimiento que siento hacia mi querida madre, quien ha estado conmigo en todo momento de dificultad. Sin ella seguramente no habría llegado hasta aquí y el terminar esta carrera universitaria es algo que no solo me debo a mi mismo, sino que siento le debo a mi madre, y a mi padre también por no tirar jamás la toalla conmigo, y por su apoyo constante pese a la distancia. ¡Gracias mamá! ¡Gracias papá! Te amo mamá. Te amo papá.

A mi amada Acsa, mi mejor amiga, mi compañera de vida... por estar siempre ahí desde que te conozco. Te dedico estas palabras de Sabina: "...Y morirme contigo si te matas y matarme contigo si te mueres, porque el amor cuando no muere mata, porque amores que matan nunca mueren...".

A mi tutor Alejandro muchas, muchísimas gracias, no solo por darme un poco de tu tiempo, sabiduría, y tu maravilloso sentido del humor, sino también por regalarme un trocito de amistad. ¡Gracias Alex! Eres un crack.

Finalmente quiero agradecer a Jesús con el siguiente versículo:

¡Alaben al Señor porque Él es bueno,

y su gran amor perdura para siempre!

1 Crónicas 16:34

Resumen

La parte esencial de la recopilación de información para el apoyo a la toma de decisiones, ha sido siempre no solo identificar lo que piensan los demás sino también, y aún más importante, el estado emocional de los mismos. Esto resulta especialmente útil cuando se necesita saber cómo se siente el cliente con respecto al producto o servicio que se le brinda. Con la creciente disponibilidad y popularidad de los recursos de opinión tales como los sitios web, los blogs personales, y las redes sociales, emanan nuevas oportunidades de negocio y desafíos sociales, debido al uso intensivo de tecnologías de la información para buscar y entender las opiniones de las personas. El brote súbito de actividad en el área de la minería de opiniones y análisis de los sentimientos, que se ocupa del tratamiento computacional de la opinión, el sentimiento y la subjetividad en el texto, se ha producido como una respuesta a la demanda de nuevos sistemas informáticos orientados al análisis de las opiniones.

En nuestra Universidad se realiza al final de cada curso una encuesta por cada asignatura. Dicha encuesta no permite obtener una retroalimentación continua del proceso de enseñanza en tiempo real. Sin tener hasta fin de curso los resultados de la encuesta, no es posible rectificar hasta el siguiente curso los problemas detectados. ¿No sería mucho mejor poder detectar lo antes posible los problemas a través de la opinión de los estudiantes con respecto al curso?

Por esta razón surge mi clasificador TBONTB (alusión a la frase de Shakespeare "***T**o **B**e **O**r **N**ot **T**o **B**e*"), con el objetivo de medir a partir de una lista de palabras afectivas, la polaridad emocional de la opinión de los estudiantes. Pese a haber sido esta la motivación inicial del proyecto, el algoritmo es totalmente independiente del contexto social sobre el cual se aplica, lo que lo hace más genérico y por tanto aplicable a más campos.

Cabe señalar que además del algoritmo, se desarrollaron varias herramientas de software para evaluar la utilidad del algoritmo para los fines con que fue creado, obteniendo resultados tan prometedores que sobrepasan toda expectativa real que tenía al comienzo.

La metodología empleada para estos fines, ha sido experimentar con el algoritmo y grandes conjuntos de entrenamiento de textos evaluados positiva o negativamente en formato *microblogging* y *blogging*. De esta manera, se evaluó comparativamente las clasificaciones obtenidas por el algoritmo con las asignadas por los humanos que crearon dichos conjuntos de textos clasificados.

Producto de esta evaluación exhaustiva se refinó el umbral de decisión de polaridad emocional del algoritmo para textos en formato *microblogging* y *blogging,* hasta llegar a un 84% y 59% de acierto respectivamente.

Contenido

Índice tablas

Índice Imágenes

1.0. Introducción

La finalidad de este primer capítulo es brindar al lector una visión global de este proyecto, explicando su motivación, contexto, los objetivos planteados y cómo conseguirlos en cierto modo. También se explica la estructura del resto del documento.

En las asignaturas en la UC3M [1] se hacen encuestas al final de cada curso para saber cómo ha sido el desenvolvimiento académico tanto de los profesores como de los alumnos. Estaría bien complementar esta idea con un sistema de análisis automatizado de opiniones de los alumnos más continuo (no tan puntual como una encuesta) para saber cómo se está desarrollando el curso.

1.1. Motivación

La primera motivación por la cual surgió el clasificador de opiniones fue para explorar la efectividad de un algoritmo de clasificación de opiniones, que fuese capaz de determinar la polaridad emocional de una opinión humana, en este caso, de los alumnos de la universidad, para ser utilizado en un futuro en los foros de las asignaturas, con el propósito de ser capaces de detectar problemas durante el curso académico en tiempo real; por ejemplo, cuellos de botella que surgen cuando coinciden muchas actividades docentes de distintas asignaturas. Pero esta no es la única motivación. En las siguientes líneas se dan a conocer otras motivaciones también muy importantes.

Para hacerse una idea de la cantidad de información que está generando en Internet el fenómeno *Big Data*, basta con escribir en Google ambas palabras para obtener nada más y nada menos que 1.720.000.000 resultados. Grande es el revuelo e interés que causa en la red este fenómeno.

Pero primero que todo, ¿Qué es el *Big Data*? Concretamente hablando, existen 4 dimensiones claves para comprender el problema, las cuales se conocen como las 4 "V" [2]:

- "Volumen" de datos
- "Velocidad" de procesamiento
- "Variedad" de datos
- "Valor" de datos

Otro punto importante para entender el *Big Data* es el término "Magnitud", la cual se refiere a la complejidad del sistema necesario para manipular los datos, y no a la dimensión del volumen de datos, como suele parecer a primera vista.

Una arista del *Big Data* es la llegada de la Web 2.0, la cual ha generado en Internet una ingente cantidad de opiniones en forums, blogs, review sites, y redes sociales como *Twitter*, *Facebook*, entre otras. Como resultado de esta tendencia, actividades que tradicionalmente se consideraban privadas, se han ido convirtiendo cada vez más en públicas. Por ejemplo, personas que anteriormente escribían un diario de su vida cotidiana, mantenían un álbum familiar de fotos y conversaban con sus amigos por teléfono o usando correo electrónico, ahora publican cada vez más su vida en blogs, postean sus fotos en sitios públicos para

compartir este tipo de contenido, y tienen conversaciones a través de muros públicos de posts en redes sociales.

En consecuencia, la web ha dado abrigo a un archivo masivo de comunicaciones humanas y de sentimientos, conteniendo grandes cantidades de valiosa información, que no es bien manipulada por los métodos tradicionales de recuperación y presentación de la información. Por ello no siempre es fácil de encontrar y analizar las opiniones necesarias para la toma de decisiones. Aquí es donde entran a jugar un papel importante las técnicas de *Social Data Mining*.

¿Por qué es necesario saber qué piensan otras personas? Saber lo que piensan otros siempre ha formado parte de la toma de decisiones en la vida diaria. Preguntas como:

- ¿Por qué votas por X?
- ¿Qué te parece el nuevo iphone de Apple?
- ¿Ganará España el próximo mundial de futbol?
- ¿Cómo se sintió Madrid cuando perdió la sede de los próximos juegos olímpicos?

Ésta y otra preguntas similares a ésta son consultas que los motores de búsqueda de Internet no manejan bien. Un caso de éxito [3] muy famoso, en el que se han aplicado métodos de análisis de sentimientos para el apoyo a la toma de decisiones, fue el manejo del *Big Data* que se realizó en la campaña electoral de Obama, ya que permitió descubrir por ejemplo que Michelle Obama era un gran reclamo para conseguir financiación en primavera.

En especial, el método empleado en este trabajo para desarrollar un clasificador de opiniones, pertenece al género de *Opinion Mining* o *Sentiment Analysis*. Por *Sentiment Análisis* [4] entendemos el estudio computacional de opiniones, sentimientos y emociones expresadas en texto. Este análisis ayuda a determinar la actitud del autor del texto con respecto a algún tópico.

Una tarea básica de *Opinion Mining* es clasificar la polaridad de un texto dado en positiva, negativa, o en algunos casos neutrales. Sistemas avanzados de clasificación de sentimientos son capaces de clasificar estados emocionales tales como el enojo, la tristeza o la felicidad.

La motivación de este trabajo es construir un método innovador de clasificación de opiniones **genérico** que se aproxime al juicio humano. Con todo ello se busca una aplicación que suponga un nuevo recurso para profesores de la universidad, ya que podría emplearse para clasificar opiniones de alumnos en los foros de las asignaturas, por ejemplo. Esto sustituiría a la encuesta que se realiza casi al final del curso, la cual es incapaz de:

1) Conocer de forma más continua (en tiempo real) la aceptación positiva a los cambios que se están haciendo a la asignatura.
2) Tomar decisiones: solucionar los problemas con la mayor prontitud posible.

1.2. Objetivos

El objetivo fundamental de este proyecto es desarrollar un método para clasificar la polaridad de opiniones en lenguaje natural, con formato de texto plano, en positiva o negativa, con una

precisión similar a la de los seres humanos. El método es libre de contexto, es decir, no es ajustado a un tópico determinado y parte del objetivo de este trabajo es demostrar que pese a ser genérico [5] obtiene muy buenos resultados.

Las funcionalidades que cumple el clasificador son:

- Evaluación de *ground truth* (conjunto de entrenamiento) en formato *microblogging*.
- Evaluación de *ground truth* (conjunto de entrenamiento) en formato *blogging*.
- La evaluación del clasificador aportará las siguientes medidas:
    - Porcentaje de acierto global.
    - Porcentaje de acierto positivo.
    - Porcentaje de acierto negativo.
    - Valencia media global.
    - Desviación estándar global.
    - Valencia media positiva.
    - Desviación estándar positiva.
    - Valencia media negativa.
    - Desviación estándar negativa.
    - Porcentaje de textos no clasificados del conjunto de entrenamiento.
- Clasificación en positivo o negativo de un texto escrito o pegado desde el portapapeles.
- Generación automática de texto aleatorio con una longitud por defecto similar a la de un *tweet*, a partir de un texto de entrenamiento.
- Clasificación en positivo o negativo de un texto generado aleatoriamente.

Otras funcionalidades deseables para el clasificador, aún no implementadas son:

- Comparación de diferentes fórmulas de clasificación basadas en los parámetros *ANEW* [6], utilizando la herramienta *Weka* [7], para obtener la mejor fórmula de clasificación posible.
- Clasificar el tópico o tema al que se refiere la opinión.
- Clasificar los sentimientos que se manifiestan en la opinión.

Con estas funcionalidades añadidas se pretende no solo clasificar en positiva o negativa una opinión, sino también saber cómo se siente el autor de la opinión clasificada con respecto al tópico referido.

1.3. Estructura del documento

El presente documento se encuentra divido en varios capítulos, recogiendo toda la información relacionada con el desarrollo de este proyecto. A continuación se ofrece un resumen del contenido de cada uno de ellos.

Capítulo 1 Introducción

En este capítulo se establece el contexto que abarca el desarrollo del proyecto, describiendo brevemente las ideas del mismo y los objetivos que se desean alcanzar, presentando motivaciones y soluciones para satisfacer las necesidades del problema planteado.

Capítulo 2 Estado de la cuestión

Durante este capítulo se describe el contexto actual en que se desarrolla el clasificador de opiniones. ¿Qué algoritmos hay para analizar sentimientos y opiniones y con qué enfoques y limitaciones? ¿Qué sistemas similares existen? Esto y más se verá en este apartado.

Capítulo 3 Análisis, Diseño, Implementación e Implantación

En este capítulo se abordan cómo evolucionaron las fases de análisis, diseño, implementación e implantación del proyecto; teniendo en cuenta todos los elementos característicos de cada fase según las necesidades del proyecto.

Capítulo 4 Evaluación

Este capítulo se refiere a qué pruebas se sometió al clasificador de opiniones del proyecto para validar los objetivos planteados inicialmente. Se hace un especial énfasis en cuán acertado es el clasificador con respecto a la polaridad emocional humana consensuada en los conjuntos de entrenamiento en formato *microblogging* y *blogging*.

Capítulo 5 Planificación y presupuesto

En este capítulo se explica la gestión del proyecto mediante la estimación del tiempo planificado para lograr los objetivos planteados en todas las fases del ciclo de vida del proyecto y el cálculo del coste total del mismo.

Capítulo 6 Conclusiones y trabajo futuro

Este apartado recoge las conclusiones extraídas a lo largo del desarrollo del proyecto, basándose en los objetivos iniciales, y los problemas encontrados durante el proceso completo de realización. A su vez se describen los conocimientos aprendidos durante el desarrollo y posibles mejoras a tener en cuenta para el futuro.

Capítulo 7 Bibliografía

Referencias a libros, documentos, webs y conceptos aparecidos a lo largo del documento.

2.0. Estado de la cuestión

2.1. Dominio de *Opinion Mining*

A menudo la disciplina de *Opinion Mining* es asociada con *Information Retrieval* (IR). A pesar de las semejanzas entre ambos dominios, *Opinion Mining* ha probado ser una tarea mucho más difícil de realizar. Ello se debe fundamentalmente a las características de las fuentes de

datos. En IR, los algoritmos de recuperación operan sobre datos objetivos, a diferencia de *Opinion Mining*, en la cual los datos de entrada solo son información subjetiva. En la práctica, esto significa que *Opinion Mining* requiere ir un paso más allá que IR y analizar oraciones y frases más profundamente con respecto a su semántica. En *Opinion Mining* la tarea fundamental es determinar la naturaleza de la opinión: cuando es positiva o negativa; qué describe; qué tópicos son valores y cuáles no, entre otras cosas.

Una de las características del contenido generado por el usuario en la Web es el desorden textual y la alta diversidad. El estilo de escritura cambia mucho dentro de un mismo portal, pero muchísimo más varía si se analiza un determinado tópico en Internet a gran escala. Tenemos el caso de la crisis económica en España, donde existe gran polémica en cuanto a las causas de su origen, pues según el contexto social en el cual se publique la opinión al respecto, unos dirán que la culpa es de los bancos, otros que el pueblo español ha vivido por encima de sus posibilidades, otros que la corrupción política y mala gestión administrativa nos ha traído a esta situación, e incluso hay quienes hablan de una crisis internacional surgida en EE.UU y que nos afecta. Además de esto, el vocabulario en términos económicos empleados varía también en gran manera para referirse al tema, términos como Ibex-35, tasa de paro, Euribor entre otros encabezan los titulares.

Las opiniones se expresan con un lenguaje informal. La gramática de construcción de oraciones puede depender mucho de la comunidad. Por ejemplo, las críticas literarias del club de fans de un libro cualquiera pueden ser totalmente diferentes e incluso inentendibles para personas de otra comunidad, como pueden ser los jugadores de un juego por ordenador inspirado en el mismo libro y viceversa.

2.1.1. Tipos de opinión

Normalmente existen dos maneras de expresar las emociones y los sentimientos: a través de opiniones directas, o comparaciones. Las opiniones directas normalmente describen un objeto y contienen adjetivos, a diferencia de las comparaciones en las cuales se mencionan más de un objeto y describe la relación entre ellos.

2.1.2. Contexto de opinión

Para extraer opiniones es necesario saber acerca de qué trata. Dependiendo de la localización del portal, la información descriptiva puede ser obtenida de diversas formas. Por ejemplo, en portales de crítica o revistas, de manera general, es relativamente fácil extraer la información de los sentimientos. Sin embargo, en un forum es considerablemente más difícil identificar el sujeto de discusión de cualquier post.

Es por ello, que las aplicaciones que realizan algún tipo de análisis de sentimientos, suelen estar enfocadas a un contexto específico. Se espera entonces que una aplicación genérica funcione mucho peor que una *ad hoc*. El trabajo que nos ocupa en este proyecto pretende demostrar, que esto no es cierto del todo, pues nuestro método de clasificación es totalmente genérico supervisado por humanos y se han obtenido muy buenos resultados como se verá más adelante.

2.1.3. Nivel de interés

Mientras más interesada esté la persona en el tema del que escribe, mayor cantidad de detalles expresará en su opinión al respecto. Este factor desempeña un papel vital para la clasificación de la opinión, pues mientras más detallada sea la opinión, más factible será la clasificación de la misma.

2.1.4. Vocabulario

Las opiniones pueden usar un vocabulario explícito o implícito para expresarse. Por ejemplo, una opinión explícita como "The product is good" es más fácil de analizar y clasificar que las implícitas "I love the way this app works!" o "I was stunned to see all those special effects on the movie", las cuales contienen muchos sentimientos y son más difíciles de reconocer y clasificar.

2.1.5. *Sentiment Analysis* a nivel de documento

Este tipo de análisis intenta clasificar de manera global todos los sentimientos expresados por el autor en el texto del documento. Un documento puede ser un post de un blog o simplemente un artículo de una revista. La tarea consiste en decidir cuándo el documento es positivo, negativo o neutral con respecto a un determinado objeto. Cuando se aplica a un solo tipo de texto, estas técnicas suelen tener un rango de precisión entre el 70% y 80%, dependiendo de la cantidad de entrada humana y el tipo de texto.

El trabajo realizado por Turney [8] en clasificación de críticas presenta un enfoque basado en medir la distancia de los adjetivos encontrados en el texto de palabras preseleccionadas con una polaridad conocida (por ejemplo "excellent" y "poor"). El autor presenta un algoritmo de tres pasos que procesa los documentos sin supervisión humana. Primero, los adjetivos son extraídos junto con una palabra que provee información contextual. Las palabras a extraer son identificadas por aplicar un patrón predefinido (por ejemplo: adjetivo-sustantivo o adverbio-sustantivo). Después, la orientación semántica es medida. Esto es hecho midiendo la polaridad de palabras de polaridad conocida. La dependencia mutua entre dos palabras es encontrada por análisis de cantidad de éxitos con el motor de búsqueda de Alta Vista para documentos que contienen dos palabras coocurrentes en una cierta proximidad mutua. Al final el algoritmo calcula la media de la orientación semántica para todos los pares de palabras y clasifica la crítica como recomendada o no, según este valor y su umbral de decisión.

En cambio Pang [9] su trabajo consiste en técnicas clásicas de clasificación de tópicos. El enfoque propuesto sirve para determinar cuando un conjunto de algoritmos de aprendizaje automático pueden producir buenos resultados cuando *Sentiment Análisis* es percibido como *Document Topic Análisis* con dos tópicos: positivo y negativo. Los autores presentan resultados de experimentos con: *Naive Bayes* [10], *Máximum Entropy* [11], y *Support Vector Machine* [12]. Asombrosamente los resultados obtenidos son comparables a otras soluciones en un rango del 71% al 85% dependiendo del método y de los *tests data sets*.

2.1.6. *Sentiment Analysis* a nivel de oración

El análisis de sentimiento a nivel de oración es una acción que se asocia a dos tareas. La tarea inicial trata de identificar cuando la oración es subjetiva (opinable) u objetiva. La segunda tarea es determinar la polaridad positiva, negativa o neutral de la oración en cuestión, generalmente

solo las oraciones clasificadas como subjetivas. Similar al nivel de análisis de sentimiento a nivel de documento, se emplean técnicas de aprendizaje automático.

Riloff y Wiebe [13] depositaron el mayor impacto de su trabajo en identificar oraciones subjetivas. Ellos propusieron un método que en el arranque usa clasificadores de alta precisión para extraer un número de oraciones subjetivas. Durante esta fase las oraciones son etiquetadas por dos clasificadores: primero como alta confianza oración subjetiva y segundo como alta confianza oración objetiva. Las oraciones que no son clasificadas en ninguna de esas categorías, son dejadas sin etiquetar y omitidas en esta primera fase. Ambos clasificadores se basan en una lista previa de palabras que indican la subjetividad de la oración. El clasificador subjetivo busca la presencia de las palabras en la lista dentro de la oración, mientras que el clasificador objetivo encuentra oraciones sin estas palabras. De acuerdo con los resultados presentados por los autores, sus clasificadores tienen una precisión del 90% durante los tests.

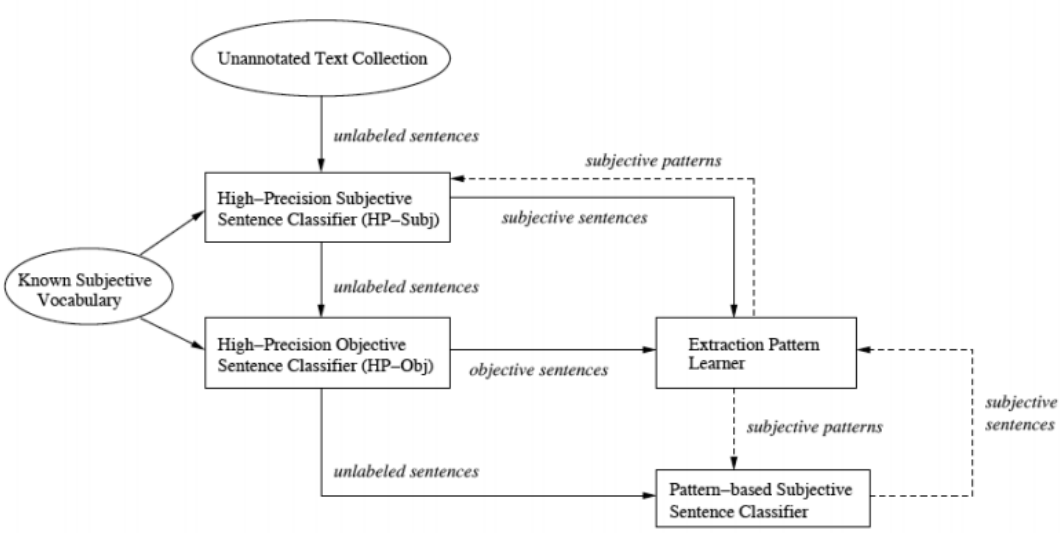

Figura 1 Fase de arranque [13]

En la segunda fase los datos obtenidos se usan para entrenar un algoritmo de extracción que genera patrones para oraciones subjetivas. Los patrones son usados para extraer más oraciones en el mismo texto. Este método intenta aumentar la exhaustividad de la recuperación de oraciones subjetivas de la primera fase. Sin embargo, tal y como se esperaba, el autor señala que la precisión descendió entre 70-80%.

SYNTACTIC FORM	EXAMPLE PATTERN
<subj> passive-verb	<subj> was satisfied
<subj> active-verb	<subj> complained
<subj> active-verb dobj	<subj> dealt blow
<subj> verb infinitive	<subj> appear to be
<subj> aux noun	<subj> has position
active-verb <dobj>	endorsed <dobj>
infinitive <dobj>	to condemn <dobj>
verb infinitive <dobj>	get to know <dobj>
noun aux <dobj>	fact is <dobj>
noun prep <np>	opinion on <np>
active-verb prep <np>	agrees with <np>
passive-verb prep <np>	was worried about <np>
infinitive prep <np>	to resort to <np>

Figura 2 Patrones de extracción. Fase de aprendizaje. Plantillas sintácticas y sus correspondientes patrones de extracción [12]

Durante la fase de aprendizaje el algoritmo emplea un conjunto predefinido de plantillas sintácticas que machean las oraciones subjetivas (ver Fig. 2). Después de todo, el conjunto de entrenamiento es procesado, los patrones extraídos son ordenados por la frecuencia de ocurrencia de los mismos y, de acuerdo a ciertas condiciones iniciales, solo los mejores patrones son elegidos para la siguiente iteración del análisis del texto (ver Fig. 3).

PATTERN	FREQ	%SUBJ
<subj> was asked	11	100%
<subj> asked	128	63%
<subj> is talk	5	100%
talk of <np>	10	90%
<subj> will talk	28	71%
<subj> put an end	10	90%
<subj> put	187	67%
<subj> is going to be	11	82%
<subj> is going	182	67%
was expected from <np>	5	100%
<subj> was expected	45	42%
<subj> is fact	38	100%
fact is <dobj>	12	100%

Figura 3 Patrones de extracción en la fase de aprendizaje – frecuencia de patrones dentro de oraciones subjetivas [12]

Aunque el presente trabajo obtuvo muy buenos resultados, solo la segunda tarea de clasificación de oraciones subjetivas en positivas, negativas o neutrales es de *Sentiment Analysis*, a diferencia del trabajo desarrollado por Yu y Hatzivassiloglou [14] trata sobre la clasificación tanto de oraciones subjetivas como las objetivas y su orientación positiva, negativa o neutral. Para el primer paso de clasificación de oraciones, los autores presentaron los resultados de tres algoritmos distintos: *sentence similarity detection, naïve Bayens classification* y *Multiple naïve Bayens classification*. En el segundo paso de reconocimiento de orientación de la polaridad de la oración los autores, usaron técnicas similares a las de Turney [8] para análisis de sentimiento a nivel de documento. La principal diferencia es que el algoritmo emplea más de dos palabras base ("excellent/poor") para comparar las palabras de la oración.

2.1.7. Sentiment Analysis a nivel de característica

El análisis de sentimiento a nivel de característica es el estudio más detallado del texto, y por tanto el más difícil de realizar, así como también el más útil. El objetivo no es solo determinar la subjetividad del texto y la polaridad sino también qué cuestión en particular le gustaba o no al autor acerca del objeto. Típicamente este objetivo es dividido en las siguientes tareas:

- Extraer características del objeto que es comentado.
- Determinar la orientación de las opiniones (positiva/negativa/neutral).
- Agrupar características que son sinónimos y producir un resumen.

Similarmente a los dos niveles descritos anteriormente, con frecuencia los experimentos de *Sentiment Analysis* a nivel de característica son enfocados a un tipo de texto. Algunos autores aún van más lejos y presentan métodos para un formato de texto específico, por ejemplo revisiones en las que las características positivas y negativas son separadas en áreas diferentes. Tal enfoque es presentado por Hu y Liu en su trabajo acerca de *customer review analysis* [15]. En su investigación, los autores presentan *opinion mining* basada en la frecuencia de sus características. Solo las características más frecuentes son tenidas en cuenta para generar el resumen.

2.2. ¿Qué hay similar?

Dado que mi método de clasificación emplea una lista de palabras afectivas *Affective Norms for English Words* (*ANEW*) [16], para medir la valencia psicológica media de un texto, en una escala del 1 al 9, con umbral de decisión 5 a continuación se mencionan otras listas de palabras afectivas que se emplean en *Opinion Mining*:

- *ANEW* [17]. Una lista de palabras construidas por Bradley y Lang [17] de 1034 palabras tasadas para valencia, excitación y dominación. Tiene la restricción de que solo puede ser usada para fines académicos.
- *AFINN* [18]. Una lista de palabras creada por Finn Nielsen [18] tasadas con valencia por un entero entre menos cinco y más cinco.
- *EmoLex* [19]. Una larga lista de más de 24 000 palabras.
- *LabMT* [20]. Una larga lista de palabras que contiene entre otras medidas el *twitter rank* y *google rank*.

- *WordNet-Affect* [21]. Una larga lista de palabras, que es parte de WordNet Domains.

Estas son algunas de las más importantes que he encontrado y todas son *free software*.

A continuación una lista de herramientas que se emplean para *Opinion Mining*:

- *SentiStrength* [22]. Es una herramienta que estima la orientación positiva o negativa de textos cortos, incluso en lenguaje informal.
- *Pattern* [23]. Es una libería de python para text mining, que contiene herramientas de procesamiento de lenguaje natural y algoritmos de aprendizaje automático, e incluye *Sentiment Analysis*.
- *Sasa-tool* [24]. Es una herramienta enfocada en *Sentiment Analysis open-source*.
- *Senti by Crowflower* [25]. Es una herramienta comercial para *Sentiment Analysis*.

Un artículo en el cual se puede profundizar en las herramientas *open-source* de *Sentiment Analysis* actualmente es el de Seth Grimes [26].

A continuación una lista de servicios online para *Sentiment Analysis*:

- http://sentimentalytics.com/ . Un plugin de navegador que automáticamente analiza contenido de redes sociales.
- http://www.sentiment140.com/ . Permite saber cuál es el sentimiento de *Twitter* relacionado con una marca o producto determinado.

2.2.1. *We feel fine*

We feel fine [27] fue mi primer contacto con el mundo del análisis de sentimientos. Es un motor de búsqueda emocional online que contiene 2.178 sentimientos de lengua inglesa registrados [28], el cual siempre está escaneando *blogs, microblogs* y varias redes sociales, extrayendo oraciones que contengan las palabras "*I feel*" o "*I am feeling*", así como el sentimiento (si está explícito), género, edad, sexo y otros datos del autor y del post. Para consultar la base de datos se emplea una API [29] que a través de http requests, con parámetros predefinidos en la url, devuelve en el http response un documento xml, que contiene como máximo 1.500 post para un mismo día. Siempre devolverá los mismos 1.500 para un mismo día, aunque en la base de datos existan muchos más registros (Ver Figura 4).

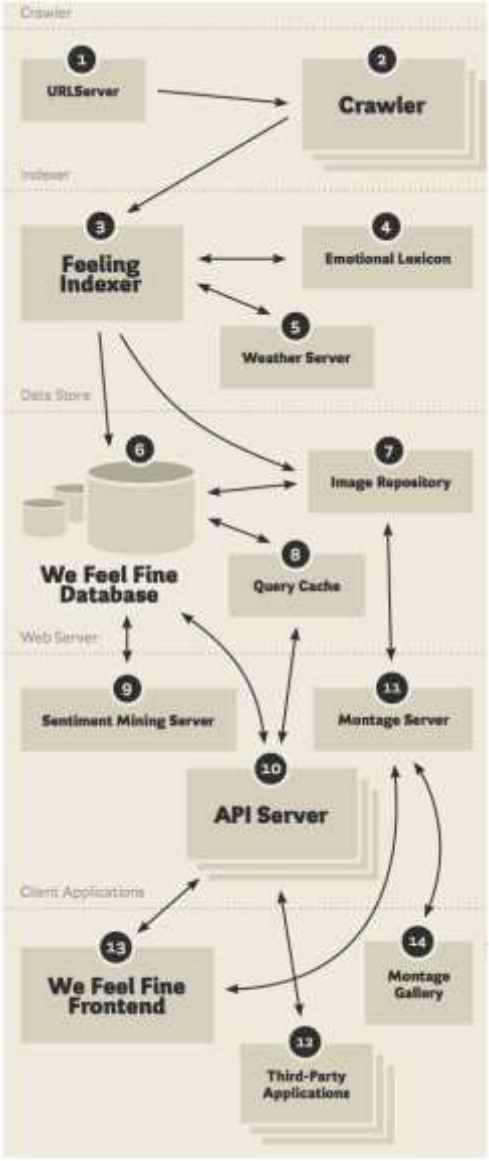

Figura 4 Diagrama de componentes de We Feel Fine [30]

La similitud de *We Feel Fine* con mi clasificador, es que cuando el sentimiento está expresado de manera explícita, como por ejemplo "*I feel bored*" en el elemento xml asociado a esta oración existirá un atributo *feeling*="*bored*", porque "*bored*" pertenece a la lista de 2178 sentimientos mencionada, lo cual, es de por sí clasificar una oración de naturaleza subjetiva debido a la conjugación del verbo sentir, por la característica emocional que expresa el autor. Es decir, es *sentiment analysis* a nivel de característica, enfocado totalmente a la característica emocional.

A pesar de que mi clasificador, es a nivel de documento y oración el análisis que realiza (y no a nivel de característica como *We Feel Fine*), es totalmente viable este tipo de análisis en mi clasificador como se verá en el apartado de trabajos futuros, propongo una solución para cumplimentar esta capacidad en mi clasificador y ello es a mi juicio una similitud en potencia.

A diferencia de mi clasificador, We Feel Fine no clasifica la polaridad de las oraciones que almacena. De hecho, hay oraciones en las que el sentimiento está expresado implícitamente debido a la alta carga de subjetividad de la misma, pese a la presencia de la conjugación del verbo sentir. Por lo tanto se omite el sentimiento para estos casos.

2.2.2. Hedonometer

Hedonometer es un sistema que recolecta información de *Twitter* en inglés para evaluar y determinar el nivel de felicidad de personas, basado en los datos online que ellos producen. El objetivo de los creadores de este proyecto: Peter Dodds [31] y Chris Danforth [31] en la universidad de *Vermont's Computational Story Lab*, no fue solo medir la felicidad humana sino también hacerlo en tiempo real.

Para ello, el hedonómetro analiza expresiones online de personas en redes sociales (ciertamente un terreno rico en este tipo de material emocional), con un enfoque similar al del equipo de *We Feel Fine*. En la primera versión de la plataforma solo usaron *Twitter* como fuente de datos, pero los diseñadores de la misma, dicen que puede ser expandida a cualquier fuente de datos. Por ahora, el inglés es el único lenguaje soportado, embargo se desea añadir más lenguajes, así como una API de consulta en el futuro.

Para cuantificar la felicidad basada en el lenguaje, el hedonómetro mezcló las 5.000 palabras más comunes de Google *Books* [32]*, New York Times articles* [33]*, Music Lyrics* [34], y *Twitter* [35]. El resultado de esto fue un conjunto compuesto de alrededor de 10.000 palabras en inglés únicas. El sistema de puntuación de estas palabras varía en una escala de *sad* (1) hasta *happy* (9), el cual se puede comprobar en la web [36], donde se muestran varias medidas de las cuales, la que hacemos referencia en este apartado es la llamada *hapyness average*. Este sistema de puntuación está basado en *Amazon's Mechanical Turk service* [37], el cual emplea inteligencia humana para desarrollar tareas que los ordenadores no son capaces de realizar.

Para ilustrar el uso de este sensor de felicidad en tiempo real, mostraremos la evaluación histórica del hedonómetro en dos días señalados cuya polaridad emocional es opuesta en el contexto social de los Estados Unidos de Norteamérica. Veamos qué midió el 15 de abril del 2013, el día del atentado de la maratón de Boston, un día ciertamente triste:

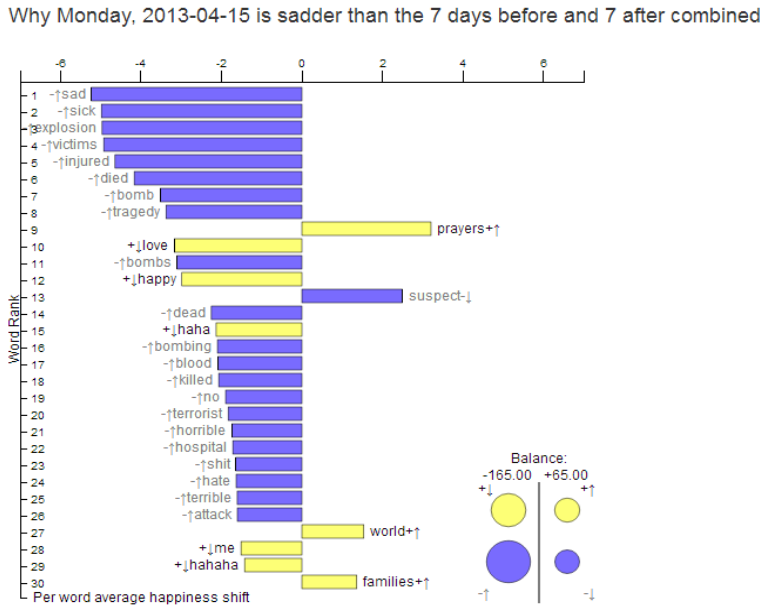

Figura 5 Evaluación del hedonómetro el 15 de abril del 2013

Como se puede constatar en la Figura 5, fue un día triste. Las palabras que más relevancia tuvieron fueron *sad, sick, explosion, victims* etc. Esto corrobora que el hedonómetro lo haya medido como uno de los días más tristes.

En cambio, el día de acción de gracias del 25 de Diciembre del 2008 fue uno de los más felices como se muestra en la siguiente imagen:

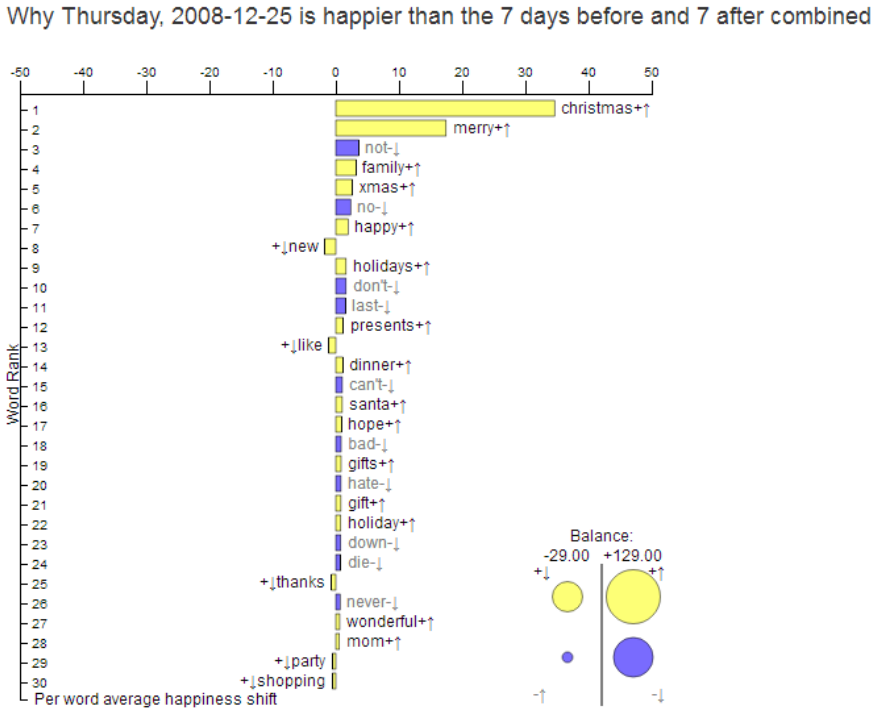

Figura 6 Evaluación del hedonómetro el 25 de Diciembre del 2008

Para que veáis una imagen del histórico [38] de felicidad registrado por el hedonómetro desde el 2008 hasta la actualidad:

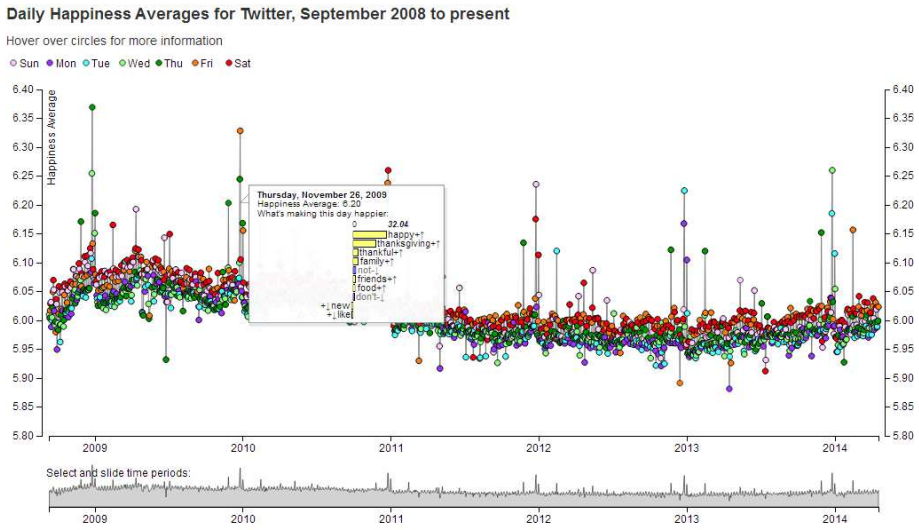

Figura 7 Hedonómetro completo

A estas alturas sería correcto preguntarnos entonces: ¿En qué consiste la similitud entre mi clasificador y el hedonómetro? Básicamente consiste en dos similitudes. La primera de ellas es que para cuantificar la polaridad del estado emocional de un texto empleamos la misma fórmula que emplean para evaluar textos en gran escala en su hedonómetro [39]. La segunda es que emplean una lista de palabras afectivas ANEW [17] al igual que mi clasificador, aunque no es la misma fuente de datos.

Una vez vistas las semejanzas, ¿Cuáles son las diferencias en el hedonómetro y mi propuesta? Una de las diferencias más notables a simple vista, es el formato de clasificación de la polaridad, que en el hedonómetro es un valor real en el intervalo [1; 9], a diferencia de mi clasificador, el cual obtiene también un valor real dentro de el mismo intervalo, pero lo nominaliza empleando un umbral de decisión que depende del formato del texto (microblogging o blogging), haciendo corresponder positivo a los valores mayores o iguales al umbral y negativo a los inferiores. También es distinto en cuanto al nivel de granulidad, puesto que el hedonómetro está pensado para realizar análisis de sentimientos a nivel de día, donde cada día es una recolección de todos los *tweets* registrados por hedonómeter para la fecha correspondiente, mientras que el clasificador que aquí se presenta analiza sentimientos a nivel de documento y de oración. Otra diferencia significativa es que el hedonómetro al tener como origen de datos *Twitter* es dependiente del contexto social relacionado con esta red social a nivel mundial, pero sobre todo en Estados Unidos de Norteamérica (por este motivo se le ha llamado sensor de la felicidad en Norteamérica); en cambio mi clasificador es totalmente independiente del contexto social.

2.3. ¿Qué elementos has usado?

Los puntos que se tratarán en los siguientes apartados son los elementos usados para la construcción del clasificador. Ellos son una lista de palabras afectivas y una fórmula de estimación de valencia media de las mismas.

2.3.1. Lista de palabras afectivas

La lista de palabras afectivas que empleo fue descargada en formato csv (*comma-separated values*) de la web http://crr.ugent.be/archives/1003 basadas en las normas ANEW [17]. Me decidí por esta colección de Amy Warriner [40] y Victor Kuperman [40] porque a diferencia de la colección de Bradley y Lang (1999) de 1.034 palabras en inglés [17], ésta contiene casi 14. 000 palabras. Por si fuera poco, esta colección está dada bajo la licencia *Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License*, lo cual es maravilloso, mientras que la colección de 1.034 palabras no tiene esta facilidad de uso. Cada palabra de la lista contiene varias medidas en una escala del 1 al 9.

Existen 3 conceptos importantes (dimensiones emocionales) a definir antes de explicar la fórmula que se emplea en mi clasificador. Son los siguientes:

- Valencia psicológica (*valence*): El grado de placer que produce el estímulo.
- Excitación psicológica (*arouse*): El grado de intensidad de la emoción producida por el estímulo.
- Dominancia psicológica (*dominance*): El grado de control experimentado por la persona bajo el efecto del estímulo.

Nosotros al igual que el equipo del hedonómetro nos decantamos por usar las tasas de valencia psicológica debido a que son mucho más consistentes que las de excitación y dominancia con respecto al estímulo, ya que estas dos últimas varían mucho más. Las tres magnitudes se miden en una escala del 1 al 9 para cada palabra de la lista mencionada en el anterior apartado. A continuación una tabla de resumen que justifica lo dicho sobre la valencia en la desviación estándar comparada [40]:

Magnitud	# de participantes	# de Observaciones	Media	Desviación estándar
Valencia	723	303.539	5,06	1,68
Excitación	745	339.323	4,21	2,30
Dominancia	845	281.735	5,18	2,16

Tabla 1 Descripción estadística de la distribución de cada dimensión [39].

Como se ve la desviación estándar de la valencia es mucho menor que la de las otras dos magnitudes.

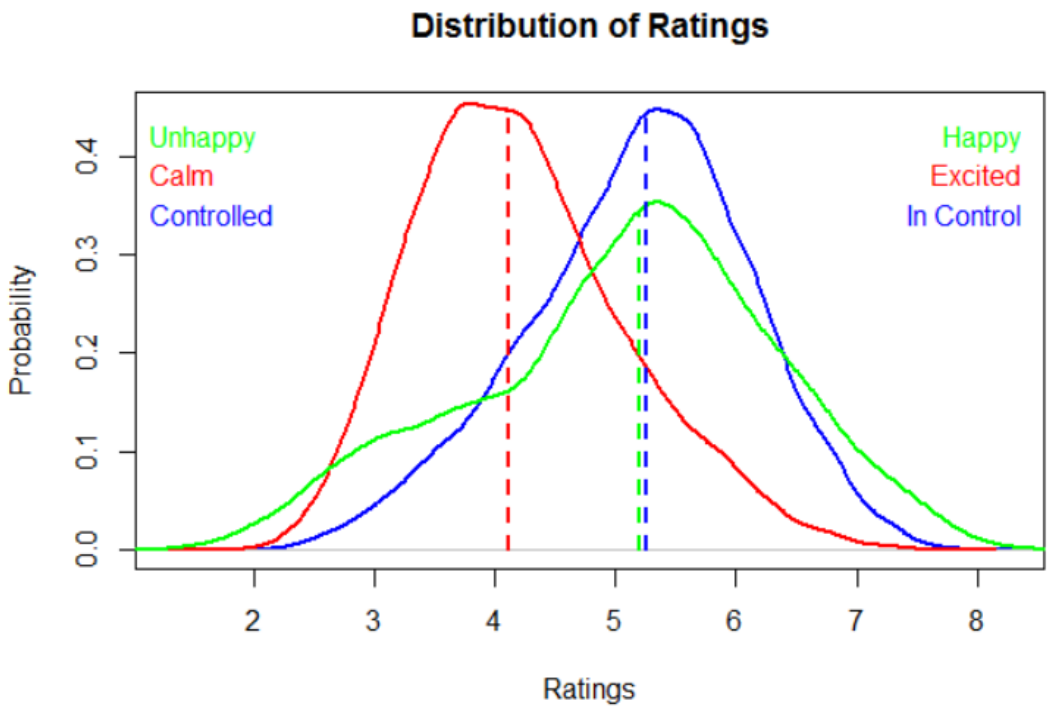

Figura 8 Distribuciones de valencia(verde), excitación(rojo) y dominancia(azul) en mi colección ANEW. Las líneas discontinuas representan las medianas de las respectivas distribuciones [39]

Es necesario destacar que cada una de estas 3 dimensiones emocionales está desglosada en 3 medidas que son media, desviación estándar y frecuencia. A su vez, cada una de estas 3 medidas de cada una de las 3 dimensiones, está ramificada en distintas sub-dimensiones, las cuales son:

- Género: M (masculino) o F (femenino).
- Edad: Y (joven) u O (viejo).
- Nivel educacional: L (bajo) o H (alto).

Para realizar este trabajo no empleamos estas sub-dimensiones de los datos de valencia, sino que simplemente usamos la valencia media genérica, es decir, de modo independiente del género, edad o nivel educacional para garantizar un clasificador de opiniones genérico. El uso de estas dimensiones se propone como parte de un trabajo futuro. Ver este apartado para mayor información al respecto.

2.3.2. Fórmula de estimación de media de la valencia psicológica en un texto

Para estimar la valencia total de un texto, la cual denotaremos como v_{text}, primero determinamos la frecuencia f_i de la i-ésima palabra de la lista de palabras afectivas ANEW que aparece en el texto, y finalmente calculamos la media ponderada de la valencia de las palabras ANEW como:

$$v_{text} = \frac{\sum_{i=1}^{n} v_i f_i}{\sum_{i=1}^{n} f_i}$$

Figura 9 Fórmula de estimación de valencia total de un texto [38]

Donde v_i es la valencia media de la i-ésima palabra del texto que aparece en la lista ANEW.

Por ejemplo, tomando la oración "'The quick brown fox jumps over the lazy dog.". Supongamos que las 3 palabras subrayadas aparecen en la lista ANEW con las valencias medias 6,64, 4,38 y 7,57 respectivamente; entonces $v_{text} = \frac{1}{3} * (1 * 6{,}64 + 1 * 4{,}38 + 1 * 7{,}57) \cong 6{,}20$. Para tener una idea más global de la escalabilidad de la fórmula veamos el siguiente ejemplo más ilustrativo:

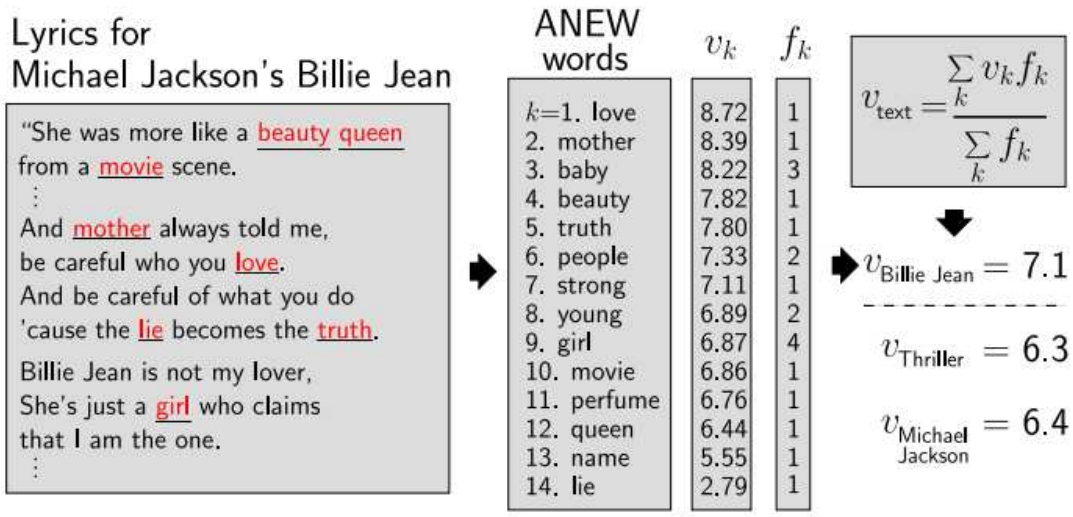

Figura 10 Valencia media de la letra de la canción Billy Jean de Michael Jackson [38]

2.3.3. Cadenas de Markov

Se conoce como cadena de Markov, a todo proceso estocástico discreto en el cual la probabilidad de que ocurra un evento depende solo del evento inmediatamente anterior. A su vez un proceso estocástico es una sucesión de variables aleatorias. Por lo tanto, una cadena de Markov es una sucesión de variables aleatorias en las que cada variable depende de la variable antecesora en la sucesión, y no de todas las anteriores.

Para la construcción de este clasificador, se desarrolló una herramienta de generación de textos aleatorios, que simulan la legibilidad humana del lenguaje natural, ya que se basa en el principio de que casi todos los escritores poseen un estilo de redacción, en el cual repiten palabras de cierto vocabulario y construyen frases con esquemas similares. Muy pocos escritores cambian su narrativa entre capítulos de una misma obra literaria. Es por ello que al hojear una novela por poner un ejemplo, si se es versado en la literatura universal se es capaz de "reconocer" el autor de la pieza en mano, sin tener todos los datos de la misma, pudiendo decir "esto parece de Cervantes o de Lorca o de Cortázar".

Teniendo en cuenta el anterior principio, se decidió usar las cadenas de Markov para generar texto en formato de oración, cuya longitud por defecto al igual que un "*tweet*" es 144

caracteres. Dichos textos generados se construían a partir de un documento de texto en lenguaje natural inglés (puesto que la lista de palabras afectivas es para lengua inglesa) del cual se parte como texto de entrenamiento.

El propósito de este generador de textos aleatorio es crear conjuntos de textos con aleatoriedad, los cuales se generarían a partir de fragmentos de textos que leídos por mí, resultarían tener una carga emocional positiva o negativa, con la premeditada pretensión de que luego los textos aleatorios generados heredasen dicha polaridad. Sin embargo, con esta técnica pronto descubrí que adolece de un problema, y es que yo soy la única persona con la que contaba para decidir, si un texto de entrenamiento para la cadena de Markov era positivo o negativo. Este hecho, me impedía realizar una evaluación realmente efectiva de mi clasificador. Por ello, decidí buscar colecciones de textos ya clasificados pero no por una sola persona sino por distintas personas como se verá en el apartado de evaluación. Es de destacar que el generador creado funciona muy bien, es independiente del lenguaje (vale para inglés, español, en fin, cualquier idioma), y de hecho, gracias a esta parte de la solución del proyecto pude comprender mejor cómo debía evaluar mi clasificador, como se verá más adelante en el apartado de evaluación.

2.3.3.1. Funcionamiento del algoritmo generador de texto aleatorio

Antes de mostrar el funcionamiento, deseo aclarar que el ejemplo utilizado en este apartado es meramente demostrativo, y con el objetivo de transmitir la idea de cómo funciona la generación de texto aleatorio de inicio a fin. Para ello se emplea un texto de partida en lengua española, aunque en mi clasificador siempre uso la lengua inglesa.

Para ilustrar el funcionamiento tomemos el siguiente texto como ejemplo:

"*El hombre armado miró enojado a su alrededor. Juan miró nervioso al hombre enjuto que rebuscaba entre las cajas, y ciertamente le miró enojado por la situación. Miró a su alrededor y le vio tan enojado y nervioso que Juan se asustó.*"

A continuación se eligen los elementos semánticamente significativos:

- Juan
- Miró
- Hombre
- Armado
- Enojado
- a su alrededor
- nervioso
- se asustó
- por la situación
- enjuto

A continuación, se muestra la matriz de trazabilidad de frecuencias de palabras consecutivas en el texto a partir de la cual se obtiene la cadena de Markov:

	Juan	miró	hombre	armado	enojado	A su alrededor	Y nervioso	Se asustó	Por la situación	Enjuto
Juan	0	0.5	0	0	0.5	0	0	0	0	0
miró	0	0	0	0	0.5	0.25	0.25	0	0	0
Hombre	0	0	0	0.5	0	0	0	0	0	0.5
armado	0	1	0	0	0	0	0	0	0	0
enojado	0	0	0	0	0	0.33	0.33	0	0.33	0
A su alrededor	1	0	0	0	0	0	0	0	0	0
Y nervioso	0.5	0	0.5	0	0	0	0	0	0	0
Se asustó	0	0	0	0	0	0	0	1	0	0
Por la situación	0	1	0	0	0	0	0	0	0	0
Enjuto	0	0	0	0	0	0	0	0	0	0

Figura 11 Matriz de frecuencias de palabras consecutivas para obtener la cadena de Markov.

El siguiente grafo, representa la cadena de Markov que se obtendría a partir del anterior texto de entrenamiento:

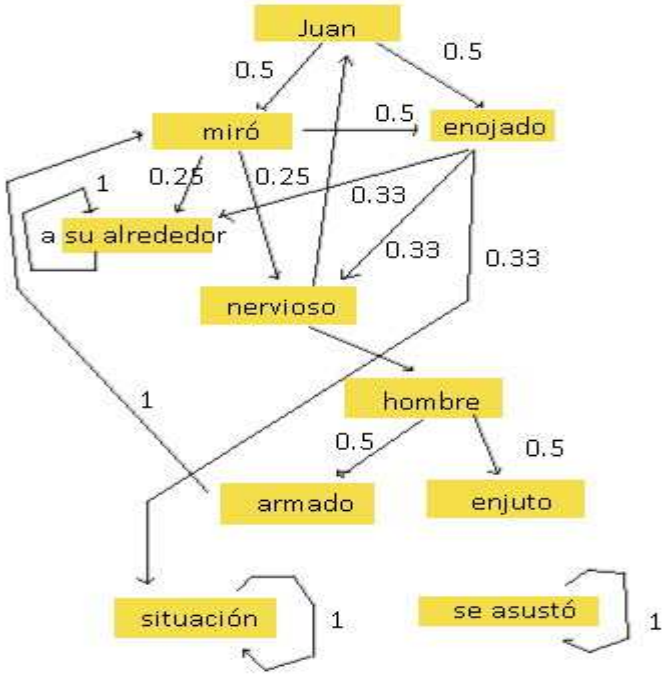

Figura 12 Grafo de la cadena de Markov.

Finalmente, el algoritmo de generación de texto aleatorio consiste en:

- Paso 1: Escoger una palabra al azar como semilla de arranque.

- Paso 2: Contar la suma total de frecuencias de las palabras consecutivas. Llamémosle N a esta suma.
- Paso 3: Generamos un número aleatorio entre 1 y N.
- Paso 4: Elegimos la palabra siguiente, basándonos en el rango de frecuencias de mayor a menor que contenga el número aleatorio anteriormente generado.
- Volvemos al paso 1, seleccionando la palabra siguiente como semilla aleatoria para la continuación de la generación del texto.
- Se termina cuando la semilla seleccionada no tiene palabras consecutivas o cuando se alcanza un límite de caracteres, por ejemplo 144.

3. Análisis, diseño, implementación e implantación

3.2. Introducción

La metodología usada ha sido ESA(*European Space Agency*) PSS-05-0 [41], la cual es el estándar de ingeniería del software aplicado en todo el software implementado por la Agencia Espacial Europea.

Las razón por la cual elegí esta metodología frente a otras como Métrica 3, o eXtreme Programming(*XP*) fue fundamentalmente porque mi clasificador de opiniones humanas es un proyecto pequeño, y para este tipo de proyectos la mejor metodología que conozco es *ESA*. En nuestro caso, los motivos por los cuales afirmo que es un proyecto pequeño son:

- El coste del desarrollo es bajo, como se verá en uno de los próximos apartados.
- Equipo de desarrollo de una sola persona.
- Solo se va a producir una versión del software.
- El tiempo planificado para realizar el proyecto ha sido menos de 6 meses, aunque como se verá en el apartado de planificación me tomó 8 meses terminar el proyecto.
- La cantidad de código fuente no excede las 10.000 líneas de código.

Métrica 3 no es apropiada para este proyecto debido a que está pensada para grandes equipos de desarrollo, y *XP* requiere al menos dos desarrolladores y una integración del equipo de desarrollo con el cliente muy frecuente. Es por ello que pensé en *ESA*.

3.3. Análisis

3.3.1. Definición del clasificador

3.3.1.1. Alcance

El clasificador de opiniones humanas, es un programa que recibe como entrada un texto en lenguaje natural y devuelve como salida la polaridad positiva o negativa del texto.

3.3.1.2. Identificación del entorno tecnológico

Para conseguir un correcto funcionamiento de la aplicación desarrollada para evaluar mi clasificador TBONTB, se requerirán los siguientes elementos tecnológicos:

- Servidor Microsoft SQL Server, versión 2005 o superior.

- .Net framwork 4.5
- Sistema Operativo [42]:
    - Windows 8.1 32 y 64 bits
    - Windows 8 32 y 64 bits
    - Windows 7 SP1 32 y 64 bits
    - Windows Vista SP2 32 y 64 bits

3.3.2. Stakeholders

El clasificador de opiniones está pensado como una herramienta para el apoyo a la toma de decisiones, de carácter genérico (es independiente del formato del texto y la comunidad parlante de la cual se extrae el texto). En especial, este clasificador está dirigido a los profesores de la universidad porque ofrece la posibilidad de obtener una retroalimentación en tiempo real del nivel de satisfacción positiva o negativa de los alumnos a través de los foros de las asignaturas. Esto es un prototipo inicial para comenzar a evaluar la efectividad del algoritmo. El clasificador también está dirigido a todo aquel interesado en el mundo del análisis de sentimientos, puesto que obtener la polaridad emocional de un texto es una de las tareas más básicas en esta disciplina.

3.3.3. Usuarios

La aplicación no requiere autentificación de usuario, por lo cual no hay distinción de roles de usuarios, pues la tarea a realizar es una sola y es común para cualquier usuario: determinar la polaridad emocional de un texto. En principio si se llega a aplicar para los foros, podrían distinguirse dos tipos de usuario, uno administrador de la aplicación y otro, el profesor.

3.3.4. Requisitos de usuario

Los requisitos recogidos para realizar la aplicación se dividen en funcionales y no funcionales. Todos los requisitos están validados. A continuación se muestran los requisitos de usuario que reflejan las capacidades que tienen los mismos para realizar en la aplicación desarrollada. Todos los requisitos de usuario extraídos, tendrán los siguientes campos:

- Identificador: Cadena alfanumérica compuesta por el prefijo RU y seguida del número del requisito.
- Tipo: Indica el tipo de requisito de usuario.
    - Funcional: Lo que el usuario puede hacer.
    - Restricción: Lo que el usuario no puede hacer.
- Descripción: Contiene una explicación más amplia del requisito.
- Entrada: La acción que debe realizar el usuario para obtener una respuesta del sistema.
- Salida: Lo que el usuario espera del sistema como respuesta a su acción.
- Prioridad: Describe la prioridad de implementación del requisito. Posibles valores: Baja, Media y Alta.

Identificador	RU01
Tipo	Capacidad.
Descripción	El usuario deberá poder seleccionar un archivo con una colección de textos en formato *microblogging* o *blogging* previamente clasificados en positivos o negativos.
Entrada	Ruta física al fichero.
Salida	Evaluación del clasificador en toda la colección de textos de entrada.
Prioridad	Alta.

Tabla 2 Requisito de usuario RU01.

Identificador	RU02
Tipo	Capacidad.
Descripción	El usuario deberá poder evaluar la efectividad del algoritmo de clasificación en función de la colección de textos de entrada previamente clasificados y seleccionados.
Entrada	Una opción de menú.
Salida	Un conjunto de medidas de evaluación.
Prioridad	Alta.

Tabla 3 Requisito de usuario RU02.

Identificador	RU03
Tipo	Capacidad.
Descripción	El usuario deberá poder elegir un texto de entrenamiento para crear la cadena de Markov.
Entrada	Ruta física al fichero.
Salida	Se muestra un mensaje de confirmación al usuario.
Prioridad	Media

Tabla 4 Requisito de usuario RU03.

Identificador	RU04
Tipo	Capacidad.
Descripción	El usuario deberá poder tokenizar el texto de entrenamiento de la cadena de Markov antes de generar el texto aleatorio.
Entrada	Opción de menú.
Salida	Se muestra un nuevo formulario con el texto de entrenamiento separado por tokens.
Prioridad	Baja

Tabla 5 Requisito de usuario RU04.

Identificador	RU05
Tipo	Capacidad.
Descripción	El usuario deberá poder evaluar el texto generado aleatoriamente por la cadena de Markov en una escala de 1 a 9.
Entrada	Texto aleatorio generado.
Salida	La media de la valencia psicológica del texto.
Prioridad	Media

Tabla 6 Requisito de usuario RU05.

Identificador	RU06
Tipo	Restricción.
Descripción	Solo se pueden generar textos aleatorios de hasta 1.000 caracteres como máximo.
Entrada	Cadena de Markov.
Salida	Texto aleatorio con 1.000 caracteres a lo sumo.
Prioridad	Baja

Tabla 7 Requisito de usuario RU06.

Identificador	RU07
Tipo	Restricción.
Descripción	Solo se dispone de una colección de textos clasificados en formato *microblogging* y en formato *blogging*.
Entrada	Ruta física al fichero.
Salida	Evaluación del algoritmo de clasificación.
Prioridad	Alta

Tabla 8 Requisito de usuario RU07.

Identificador	RU08
Tipo	Capacidad.
Descripción	El usuario podrá evaluar textos introducidos manualmente.
Entrada	Texto escrito a mano.
Salida	Evaluación del algoritmo de clasificación.
Prioridad	Alta

Tabla 9 Requisito de usuario RU08.

3.3.4.1. Matriz de dependencias de requisitos

	RU01	RU02	RU03	RU04	RU05	RU06	RU07	RU08
RU01		X					X	
RU02							X	
RU03				X	X			
RU04					X			
RU05								
RU06								
RU07								
RU08								

3.3.5. Casos de uso

En este apartado se definen los casos de uso de la aplicación desarrollada para evaluar el clasificador de opiniones.

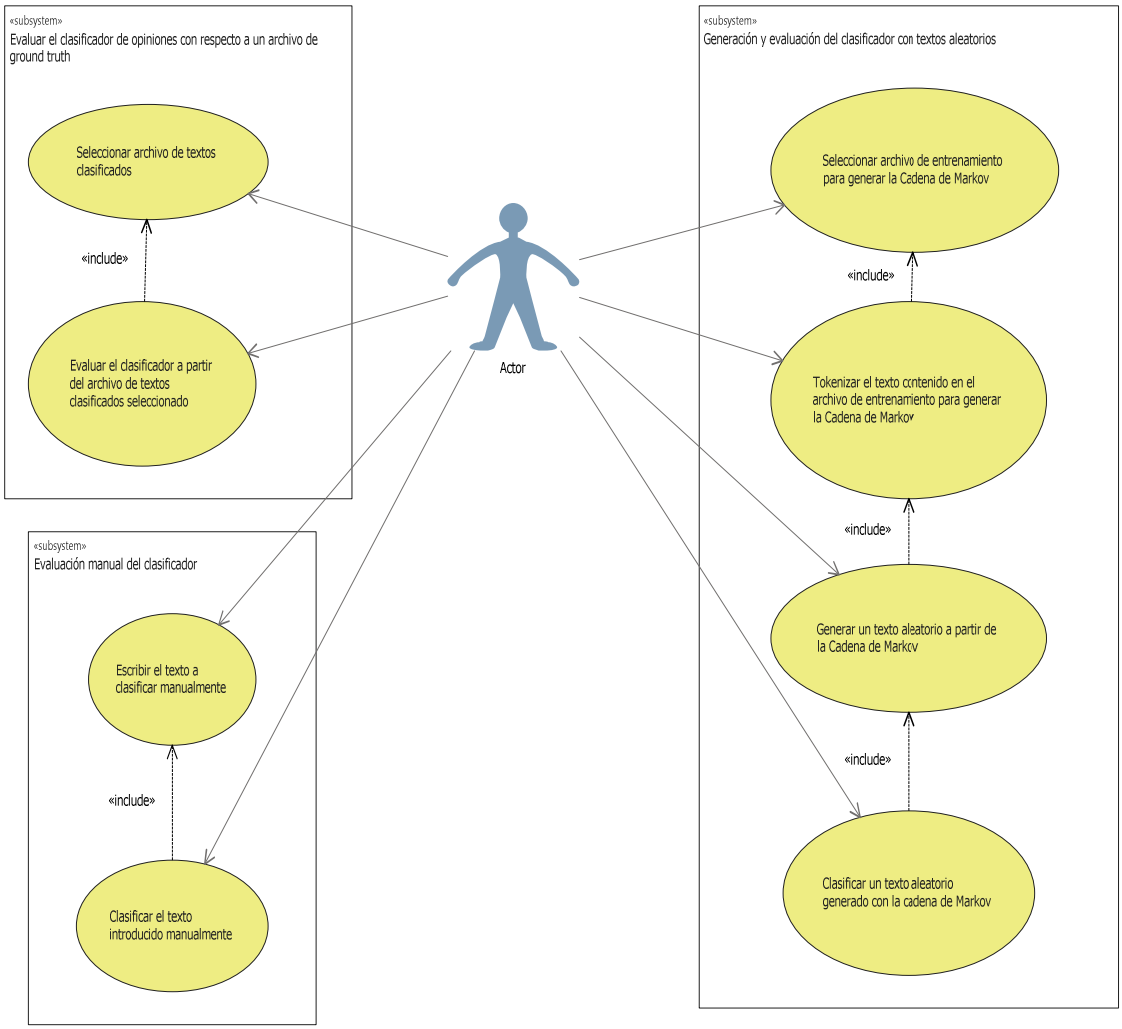

Figura 13 Diagrama uml de casos de usos.

A continuación una tabla con los casos de uso anteriores:

Identificador	Caso de uso
CU01	Seleccionar archivo de textos clasificados.
CU02	Evaluar el clasificador a partir del archivo de textos clasificados seleccionado.
CU03	Seleccionar archivo de entrenamiento para generar la Cadena de Markov.
CU04	Tokenizar el texto contenido en el archivo de entrenamiento para generar la Cadena de Markov.
CU05	Generar un texto aleatorio a partir de la Cadena de Markov.
CU06	Evaluar texto aleatorio.
CU07	Evaluar texto introducido manualmente.

3.3.6. Matriz de trazabilidad: Requisitos – Casos de usos

	CU01	CU02	CU03	CU04	CU05	CU06	CU07
RU01	X						
RU02		X					
RU03			X				
RU04				X			
RU05					X	X	
RU06							
RU07							
RU08							X

3.3.7. Requisitos del software

Los requisitos del software son una descripción completa de las funcionalidades del sistema a construir. Con el objetivo de clarificar más su definición, cada requisito de software posee las siguientes cualidades:

- Identificador: Cadena alfanumérica que comienza por el prefijo RS y continúa por el número del requisito.
- Tipo: Indica el tipo de requisito del software.
    - Funcional: Define el comportamiento del software.
    - No funcional: Define restricciones en el diseño o la implementación.
- Descripción: Una explicación más amplia del requisito.
- Entrada: Entrada del sistema. Opcional.
- Salida: Salida del sistema. Opcional.

- Prioridad: Describe la prioridad de implementación del requisito. Posibles valores: Baja, Media y Alta.

Identificador	RS01
Tipo	No funcional.
Descripción	Se requiere de .Net 4.5 instalado en el ordenador.
Entrada	Ninguna.
Salida	Ninguna.
Prioridad	Alta

Tabla 10 Requisito de software RS01.

Identificador	RS02
Tipo	No funcional.
Descripción	Se requiere de una instancia de Servidor Microsoft SQL Server, versión 2005 o superior para almacenar la base de datos.
Entrada	Ninguna.
Salida	Ninguna.
Prioridad	Alta

Tabla 11 Requisito de software RS02.

Identificador	RS03
Tipo	Funcional.
Descripción	La aplicación debe ser capaz de seleccionar el fichero con la colección de textos clasificados, ya sea en formato *microblogging* o *blogging*.
Entrada	Ruta al fichero.
Salida	Ninguna.
Prioridad	Alta

Tabla 12 Requisito de software RS03.

Identificador	RS04
Tipo	Funcional.
Descripción	La aplicación debe ser capaz de leer la lista de palabras afectivas ANEW de la base de datos.
Entrada	Conexión a la base de datos.
Salida	Lista de palabras ANEW.
Prioridad	Alta

Tabla 13 Requisito de software RS04.

Identificador	RS05
Tipo	Funcional.
Descripción	La aplicación debe mostrar los resultados de evaluar una colección de textos previamente clasificados por seres humanos.
Entrada	Ruta física al fichero.
Salida	Resultados de evaluación.
Prioridad	Alta

Tabla 14 Requisito de software RS05.

Identificador	RS06
Tipo	Funcional.
Descripción	La aplicación debe ser capaz de seleccionar el fichero de entrenamiento para crear la Cadena de Markov.
Entrada	Ruta física al fichero de entrenamiento de la Cadena de Markov.
Salida	Ninguna.
Prioridad	Media

Tabla 15 Requisito de software RS06.

Identificador	RS07
Tipo	Funcional.
Descripción	La aplicación deberá mostrar una ventana con todos los tokens detectados después de tokenizar.
Entrada	Ruta física al fichero de entrenamiento.
Salida	Tokens.
Prioridad	Baja

Tabla 16 Requisito de software RS07.

Identificador	RS08
Tipo	Funcional.
Descripción	La aplicación deberá ser capaz de generar un texto aleatorio a partir de la Cadena de Markov.
Entrada	Cadena de Markov.
Salida	Texto aleatorio relativamente legible.
Prioridad	Alta

Tabla 17 Requisito de software RS08.

Identificador	RS09
Tipo	Funcional.
Descripción	La aplicación deberá ser capaz de evaluar en una escala del 1 al 9 el texto generado aleatoriamente.
Entrada	Texto aleatorio.
Salida	Número entre 1 y 9.
Prioridad	Alta

Tabla 18 Requisito de software RS09.

Identificador	RS10
Tipo	Funcional.
Descripción	La aplicación deberá ser capaz de evaluar en una escala del 1 al 9 el texto introducido manualmente.
Entrada	Texto introducido manualmente.
Salida	Número entre 1 y 9.
Prioridad	Alta

Tabla 19 Requisito de software RS10.

3.3.8. Matriz de trazabilidad de requisitos

	RU01	RU02	RU03	RU04	RU05	RU06	RU07	RU08
RS01								
RS02								
RS03	X							
RS04								
RS05		X						
RS06			X					
RS07				X				
RS08								
RS09					X			
RS10								X

3.4. Diseño

En este apartado se lleva a cabo el desarrollo del diseño de la solución dada al problema que nos ocupa de acuerdo al análisis previamente realizado.

3.4.1. Lenguaje de programación

Para el desarrollo de la aplicación decidí utilizar C#, porque es el lenguaje nativo de la plataforma .Net de Microsoft, el cual es orientado a objetos, robusto, y entre otras ventajas el framework de .Net pone a tu disposición una amplia gama de controles gráficos para formularios de windows, que son la interfaz gráfica elegida para el desarrollo del evaluador del clasificador y la generación de cadenas de Markov. Encima de todo esto, C# es uno de los lenguajes de programación con que mejor me manejo, por no decir el que más domino y dada la necesidad de desarrollar en un corto período de tiempo el proyecto elegí este lenguaje por lo familiar que me resulta programar en el.

3.4.2. Arquitectura de módulos de la aplicación

A continuación, se muestra una imagen del diagrama uml de componentes de los módulos de la aplicación:

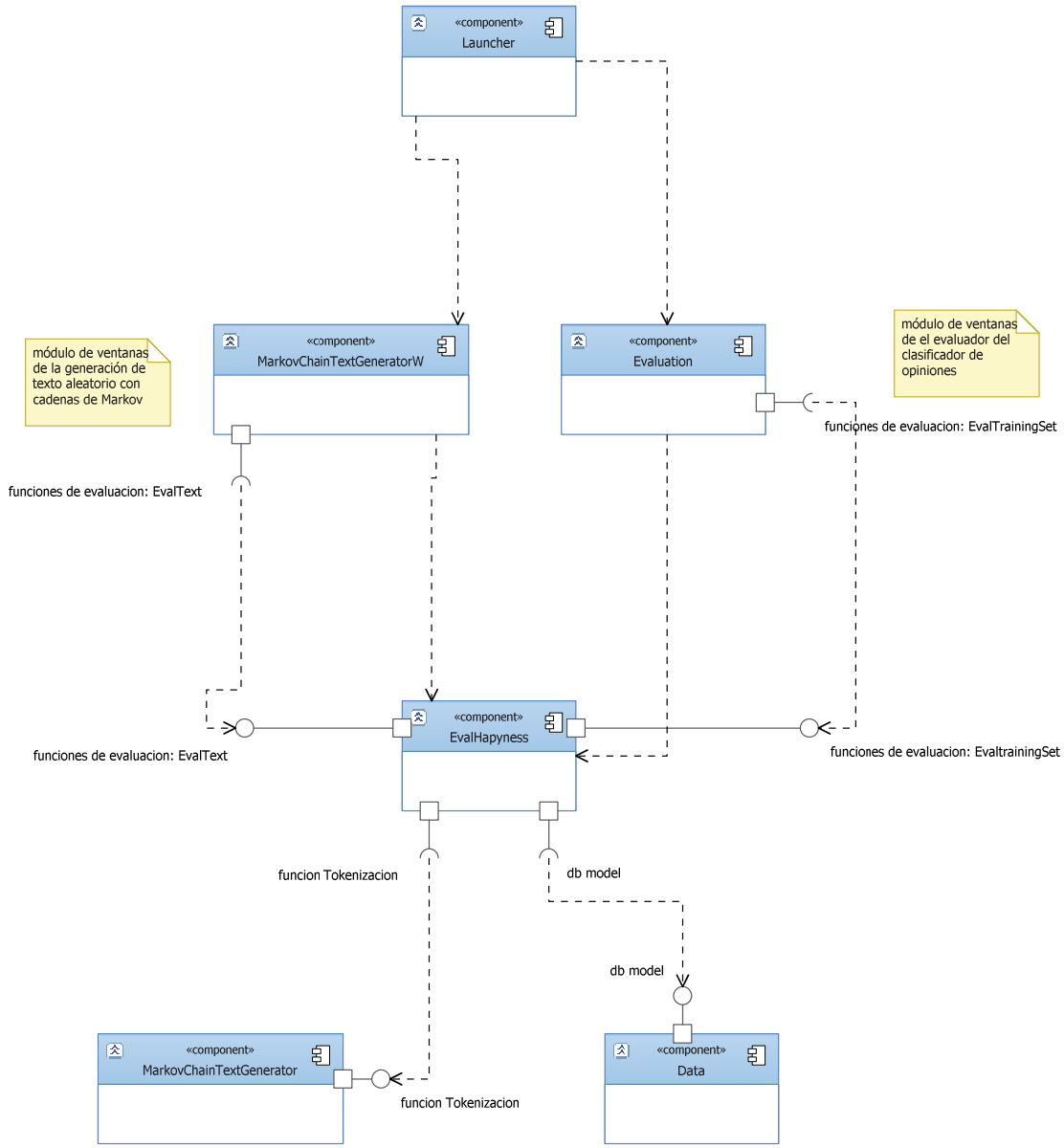

Figura 14 Diagrama uml de componentes.

Las flechas indican las dependencias entre las componentes.

3.4.2.1. Componente Launcher

La componente Launcher, es la ventana desde la cual se ejecutan el módulo principal (el evaluador del clasificador de opiniones) y el módulo secundario (el de las cadenas de Markov para generar y evaluar textos aleatorios).

3.4.2.2. Componente Evaluation

Es el componente de ventanas que ofrecen al usuario la funcionalidad de evaluar el clasificador en función de un archivo de *ground truth* (conjunto de entrenamiento), el cual puede venir en formato *microblogging* o *blogging*.

3.4.2.3. Componente MarkovChainTextGeneratorW

Es la componente de ventanas que ofrecen la funcionalidad de generar texto aleatorio humanamente legible y también permite evaluar dicho texto en una escala del 1 al 9. Igualmente, en esta componente se puede introducir texto manualmente y evaluarlo con el clasificador.

3.4.2.4. Componente EvalHapyness

Es la componente fundamental del proyecto de evaluación del clasificador. Es el corazón. Es la librería que contiene todas las funciones de evaluación.

3.4.2.5. Componente Data

La componente Data contiene el modelo entidad relacional de la base de datos.

3.4.2.6. Componente MarkovChainTextGenerator

La componente MarkovChainTextGenerator es la librería de clases y funciones donde se tokenizan los textos de entrada al clasificador de opiniones y dónde se construyen las cadenas de Markov para generar textos aleatorios.

3.4.3. Arquitectura por capas

A continuación se muestra una imagen del diagrama uml de las capas de la aplicación:

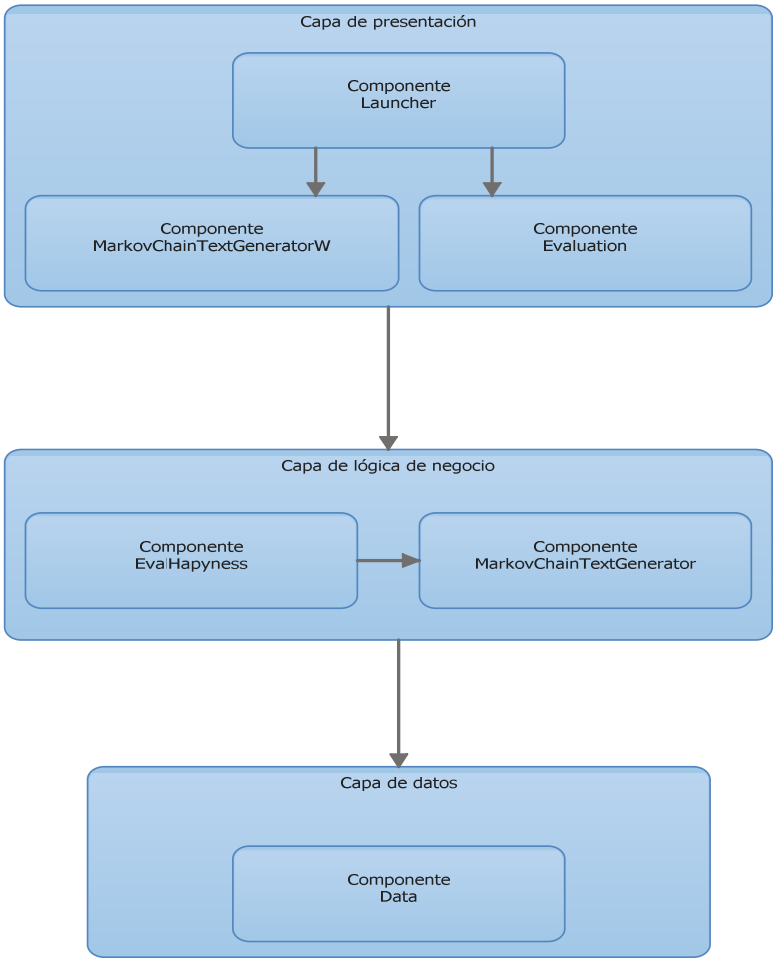

Figura 15 Diagrama uml de capas de la aplicación.

Como se aprecia en la figura 15 se distinguen 3 grandes capas:

- La capa de presentación, en la cual se ubican las componentes Launcher, MarkovChainTextGeneratorW y Evaluation. En esta capa se agrupa la interfaz gráfica de ventanas de windows de la aplicación.
- La capa de lógica del negocio, en la cual se están incluidas las componentes (Evaluation y MarkovChainTextGenerator) correspondientes a las librerías de funciones de evaluación, consultas a la base de datos y el tratamiento del texto, como es la tokenización de texto y construcción de cadenas de Markov para un texto de entrenamiento. Estas librerías aplican a los objetos instanciados las reglas del negocio del proyecto.
- La capa de datos contiene solo a la componente Data, la cual incluye el modelo de entidades relacionales de la base de datos.

3.4.4. Diagramas de clases de la aplicación

Los diagramas de clases los dividimos por las tres capas de datos, lógica de negocio y presentación. Y dentro de cada capa están separados por componentes como se verá.

3.4.4.1. Diagrama de clases de la capa de presentación

- A continuación el diagrama uml de clases en la componente Launcher:

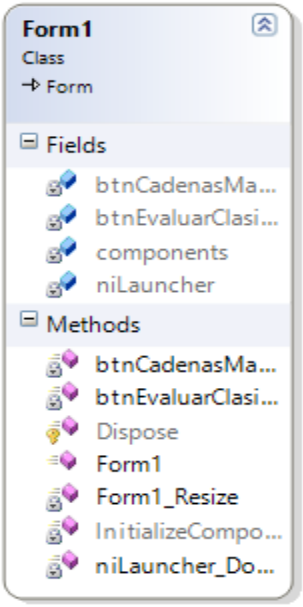

Figura 16 Diagrama uml de clases de la componente Launcher.

- A continuación el diagrama uml de clases de la componente Evaluation:

Figura 17 Diagrama uml de clases de la componente Evaluation.

- A continuación el diagrama de clases de la componente MarkovChainTextGeneratorW:

Figura 18 Diagrama uml de clases de la componente MarkovChainTextGeneratorW.

- A continuación el diagrama de clases de componente EvalHapyness:

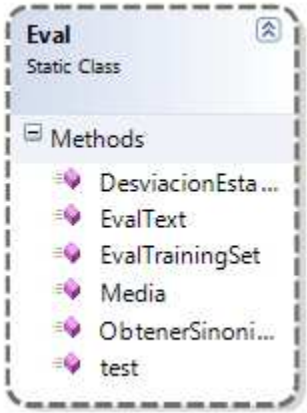

Figura 19 Diagrama uml de clases de la componente EvalHapyness.

- A continuación el diagrama de clases de la componente MarkovChainTextGenerator:

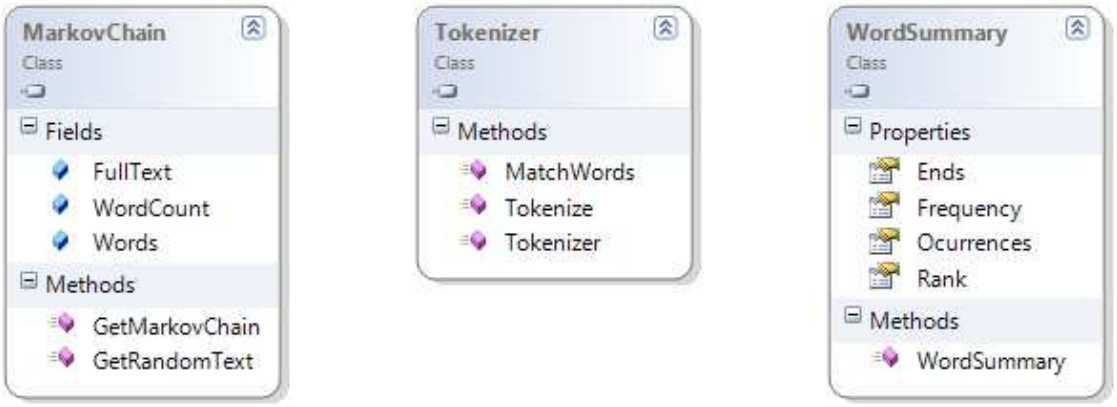

Figura 20 Diagrama uml de clases de la componente MarkovChainTextGenerator.

- A continuación el diagrama uml de clases de la componente Data:

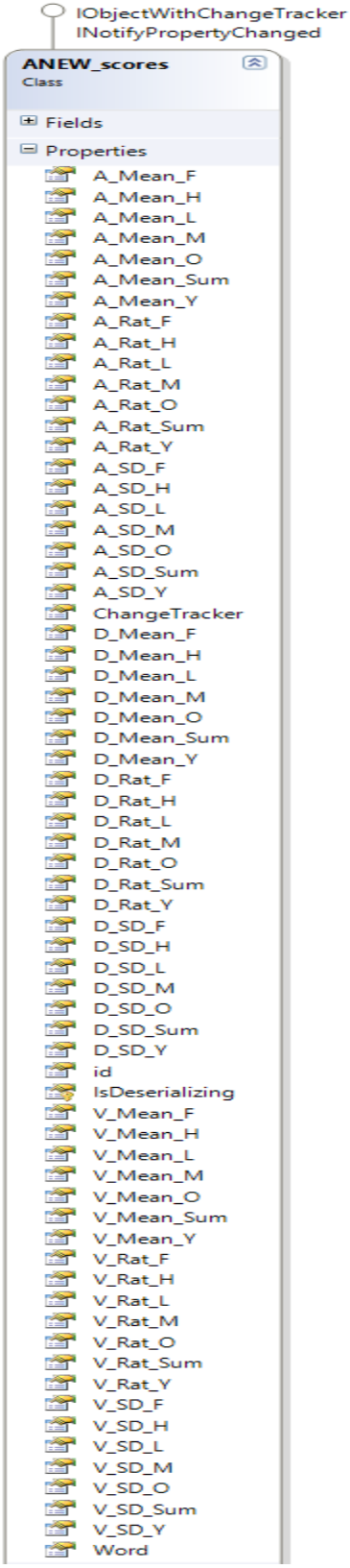

Figura 21 Diagrama uml de clases de la componente Data.

3.4.5. Modelo de la base de datos

El modelo de la base de datos solo está compuesto por una entidad que se corresponde con la tabla ANEW_scores, la cual contiene todos los datos de valencia, dominancia y arouse de cada una de las 13.914 palabras en lengua inglesa clasificadas con la siguiente definición:

Nombre de columna	Tipo de dato	Nullable
Word	varchar(50)	Unchecked
[V.Mean.Sum]	float	Unchecked
[V.SD.Sum]	float	Unchecked
[V.Rat.Sum]	float	Unchecked
[A.Mean.Sum]	float	Unchecked
[A.SD.Sum]	float	Unchecked
[A.Rat.Sum]	float	Unchecked
[D.Mean.Sum]	float	Unchecked
[D.SD.Sum]	float	Unchecked
[D.Rat.Sum]	float	Unchecked
[V.Mean.M]	float	Unchecked
[V.SD.M]	float	Unchecked
[V.Rat.M]	float	Unchecked
[V.Mean.F]	float	Unchecked
[V.SD.F]	float	Unchecked
[V.Rat.F]	float	Unchecked
[A.Mean.M]	float	Unchecked
[A.SD.M]	float	Unchecked
[A.Rat.M]	float	Unchecked
[A.Mean.F]	float	Unchecked
[A.SD.F]	float	Unchecked
[A.Rat.F]	float	Unchecked
[D.Mean.M]	float	Unchecked
[D.SD.M]	float	Unchecked
[D.Rat.M]	float	Unchecked
[D.Mean.F]	float	Unchecked
[D.SD.F]	float	Unchecked
[D.Rat.F]	float	Unchecked
[V.Mean.Y]	float	Unchecked
[V.SD.Y]	float	Unchecked
[V.Rat.Y]	float	Unchecked
[V.Mean.O]	float	Unchecked
[V.SD.O]	float	Unchecked
[V.Rat.O]	float	Unchecked
[A.Mean.Y]	float	Unchecked
[A.SD.Y]	float	Unchecked
[A.Rat.Y]	float	Unchecked

[A.Mean.O]	float	Unchecked
[A.SD.O]	float	Unchecked
[A.Rat.O]	float	Unchecked
[D.Mean.Y]	float	Unchecked
[D.SD.Y]	float	Unchecked
[D.Rat.Y]	float	Unchecked
[D.Mean.O]	float	Unchecked
[D.SD.O]	float	Unchecked
[D.Rat.O]	float	Unchecked
[V.Mean.L]	float	Unchecked
[V.SD.L]	float	Unchecked
[V.Rat.L]	float	Unchecked
[V.Mean.H]	float	Unchecked
[V.SD.H]	float	Unchecked
[V.Rat.H]	float	Unchecked
[A.Mean.L]	float	Unchecked
[A.SD.L]	float	Unchecked
[A.Rat.L]	float	Unchecked
[A.Mean.H]	float	Unchecked
[A.SD.H]	float	Unchecked
[A.Rat.H]	float	Unchecked
[D.Mean.L]	float	Unchecked
[D.SD.L]	float	Unchecked
[D.Rat.L]	float	Unchecked
[D.Mean.H]	float	Unchecked
[D.SD.H]	float	Unchecked
[D.Rat.H]	float	Unchecked

Tabla 20 Definición de la tabla de la base de datos ANEW_scores.

3.4.6. Diseño del algoritmo de clasificación

Todo algoritmo [43] se define como un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas, las cuales se realizan mediante pasos sucesivos.

La entrada de mi algoritmo es el texto de la opinión humana a clasificar.

La salida del algoritmo es la polaridad emocional de la opinión (en algunos casos de la aplicación la salida se muestra simplemente como la valencia media de la opinión). En el caso de que el texto no contenga palabras afectivas, la salida del algoritmo sería vacía.

Los siguientes pasos son el algoritmo:

- Paso 1: Tokenizar el texto. Esto es normalizar el texto eliminando concatenaciones de espacios en blancos, caracteres de tabulación, saltos de línea, retornos de carro, etc y finalmente separar el texto en *tokens*.
- Paso 2: Filtrar por palabras vacías (en lengua inglesa se conocen como *stopwords*).
- Paso 3: Hallar cuáles palabras del texto aparecen en la lista de palabras afectivas.
- Paso 4: Calcular la valencia media del texto a partir de la valencia media de cada palabra afectiva encontrada.
- Paso 5: Clasificar el texto según el umbral de decisión. El umbral de decisión aplicado para una opinión en formato *microblogging* es 5,795 mientras que para *blogging* es 5,725.

El umbral de decisión se define como el valor intermedio entre la valencia media de las opiniones cuya clase real es positiva, y la valencia media cuya clase real es negativa. Para refinar el umbral de decisión, se debe contar con conjuntos de entrenamiento de opiniones previamente clasificadas, con el fin de calcular las valencias medias de las mismas.

Vale decir, que debido al conocimiento a priori empleado, para determinar el umbral de decisión este algoritmo de clasificación pertenece a la categoría de algoritmos de aprendizaje supervisado por humanos, ya que el umbral de decisión es aprendido a partir de los conjuntos de entrenamiento de opiniones previamente clasificadas por juicio humano.

3.5. Implementación

Para abordar el tema de la implementación de la aplicación, lo haré conforme a las 3 capas definidas anteriormente en el apartado de diseño.

3.5.1. Desarrollo de la capa de Datos.

Para el desarrollo del modelo de la base de datos, que a fin de cuentas es todo lo que contiene esta capa, he empleado *Entity Framework*.

3.5.1.1. Entity Framework

Entity Framework [44] es un conjunto de tecnologías de ADO.NET que permiten el desarrollo de aplicaciones de software orientadas a datos. Los arquitectos y programadores de aplicaciones orientadas a datos, se han enfrentado a la necesidad de lograr dos objetivos muy diferentes. Deben modelar las entidades, las relaciones y la lógica de los problemas empresariales que resuelven, y también deben trabajar con los motores de datos que se usan para almacenar y recuperar los datos. Los datos pueden abarcar varios sistemas de almacenamiento, cada uno con sus propios protocolos; incluso las aplicaciones que funcionan con un único sistema de almacenamiento deben equilibrar los requisitos del sistema de almacenamiento con respecto a los requisitos de escribir un código de aplicación eficaz y fácil de mantener.

Entity Framework permite a los desarrolladores trabajar con datos en forma de objetos y propiedades específicos del dominio, como clientes y direcciones de cliente, sin tener que preocuparse por las tablas y columnas de la base de datos subyacente, donde se almacenan estos datos. Con *Entity Framework*, los desarrolladores pueden trabajar en un nivel mayor de

abstracción cuando tratan con datos, y pueden crear y mantener aplicaciones orientadas a datos.

3.5.2. Desarrollo de la capa de la lógica del negocio

El desarrollo de la capa lógica abarca dos componentes del sistema, que son Evaluation y MarkovChainTextGenerator.

3.5.2.1. Expresiones regulares

Las expresiones regulares [45] proporcionan un método eficaz y flexible para procesar texto. La notación extensiva de búsqueda de patrones coincidentes de las expresiones regulares, permite analizar rápidamente grandes cantidades de texto para buscar patrones de caracteres específicos; para validar un texto con el fin de asegurar que se corresponde con un patrón predefinido (por ejemplo, una dirección de correo electrónico); para extraer, editar, reemplazar o eliminar subcadenas de texto; y para agregar las cadenas extraídas a una colección con el fin de generar un informe. Para muchas aplicaciones que usan cadenas o analizan grandes bloques de texto, las expresiones regulares son una herramienta indispensable.

En el módulo MarkovChainTextGenerator dentro de la clase `Tokenizer` en el método MatchWords, se emplean expresiones regulares para corregir el formato del texto de entrada cuando tiene concatenación de espacios en blanco, o caracteres de tabulación, o de salto de línea dentro del texto plano de la opinión a clasificar, para finalmente devolver una colección *matches* de palabras dentro del texto tal y como se muestra a continuación:

```
public static MatchCollection MatchWords(string text)
{
      text = Regex.Replace(text, @"\s{2,}", " ");
      text = Regex.Replace(text, @"\t|\n|\r", "");
      return Regex.Matches(text, @"\b(?<word>[\p{L}0-9'\.\-
]+)\b(?<ends>[\.\?\!;:$])?");
 }
```

Figura 22 Código de método MatchWords.

3.5.2.2. Colecciones genéricas

Las clases y los métodos genéricos combinan reusabilidad, seguridad de tipos y eficacia de una manera que sus homólogos no genéricos, no pueden. Los tipos genéricos se utilizan frecuentemente con las colecciones [46] y los métodos que funcionan en ellas. La versión 2.0 de la biblioteca de clases de .NET Framework proporciona un nuevo espacio de nombres, *System.Collections.Generic*, que contiene varias clases nuevas de colección, basadas en tipos genéricos.

En el módulo MarkovChainTextGenerator, dentro de la clase `MarkovChain`, se emplea un diccionario genérico de tipo `Dictionary<string, SortedSet<string>>`, para representar la cadena de Markov como un diccionario, donde el par <key, value> significa que la llave es de tipo *string* y es una palabra cualquiera del texto, mientras que el valor asociado a esta llave es de tipo conjunto ordenado de palabras; estas son las palabras que aparecen a continuación de la palabra llave dentro del texto al menos una vez. La variable de instancia de la clase MarkovChainTextGenerator que representa el antes mencionado diccionario es como se muestra a continuación:

```
/// <summary>
/// cadena de dependencias de markov
/// </summary>
public Dictionary<string, SortedSet<string>> Words;
```

Figura 23 Colección genérica para representar una cadena de Markov.

3.5.2.3. LINQ to SQL

En *LINQ to SQL* [47], el modelo de datos de una base de datos relacional, se asigna a un modelo de objetos expresado en el lenguaje de programación del programador. Cuando la aplicación se ejecuta, *LINQ to SQL* convierte a SQL las consultas integradas en el lenguaje en el modelo de objetos y las envía a la base de datos para su ejecución. Cuando la base de datos devuelve los resultados, *LINQ to SQL* los vuelve a convertir en objetos con los que se pueda trabajar en el propio lenguaje de programación.

En el módulo EvalHapyness, en el método EvalText de la clase `Eval`, tenemos la siguiente consulta *LINQ to SQL*:

```
List<Data.ANEW_scores> lista = (from anew in model.ANEW_scores
                                //join w in words.Keys on anew.Word equals w
                                //from w in words.Keys
                                //where anew.Word == w
                                select anew).ToList<Data.ANEW_scores>();
```

Figura 24 Ejemplo de consulta *LINQ to SQL*.

3.5.2.4. Expresiones Lambda

Una expresión lambda [48] es una función anónima que se puede usar para crear tipos delegados o de árbol de expresión. Utilizando expresiones lambda, puede escribir funciones locales, que se pueden pasar como argumentos, o devolverse como valor de llamadas a funciones.

En el módulo EvalHapyness, en el método EvalTrainingSet de la clase Eval se muestra el siguiente código:

```
//obtengo las palabras que estan en ANEW
                    List<Tuple<string, double>> wordsInANEW = lista.FindAll(x =>
words.ContainsKey(x.Word.ToLower())).ConvertAll<Tuple<string, double>>(x =>
Tuple.Create<string, double>(x.Word, x.V_Mean_Sum));
```

Figura 25 Expresión lambda.

En este ejemplo, se tiene una expresión lambda que a partir de una lista devuelve una sublista que contiene todos los elementos de la lista que cumplen cierta condición; luego a partir de dicha sublista se crea otra sublista, con la misma cantidad de elementos del tipo genérico `Tuple<string, double>`. Una tupla [49] es una estructura de datos que tiene un número y una secuencia de valores concretos. La clase Tuple<T1, T2> representa una tupla de dos componentes. Una tupla de 2 componentes es similar a una estructura KeyValuePair<TKey, TValue> de un diccionario.

3.5.3. Desarrollo de la capa de presentación

El desarrollo de la capa de presentación abarca 3 componentes, las cuales son MarkovChainTextGeneratorW, Evaluation y Launcher.

3.5.3.1. Windows Forms

Windows Forms [50] es una de las dos tecnologías que se utiliza en Visual C# para crear aplicaciones cliente inteligentes, basadas en Windows que se ejecutan en .NET Framework. Windows Forms es especialmente adecuado para escenarios de desarrollo rápido de aplicaciones, donde la prioridad principal no es una interfaz gráfica de usuario compleja como en nuestra aplicación.

Como parte del desarrollo en esta capa se construyeron 3 componentes gráficas de ventanas de windows: Evaluation, MarcovChainTextGeneratorW y Launcher. A continuación se muestran algunas imágenes.

Figura 26 Ventana de bienvenida del Launcher.

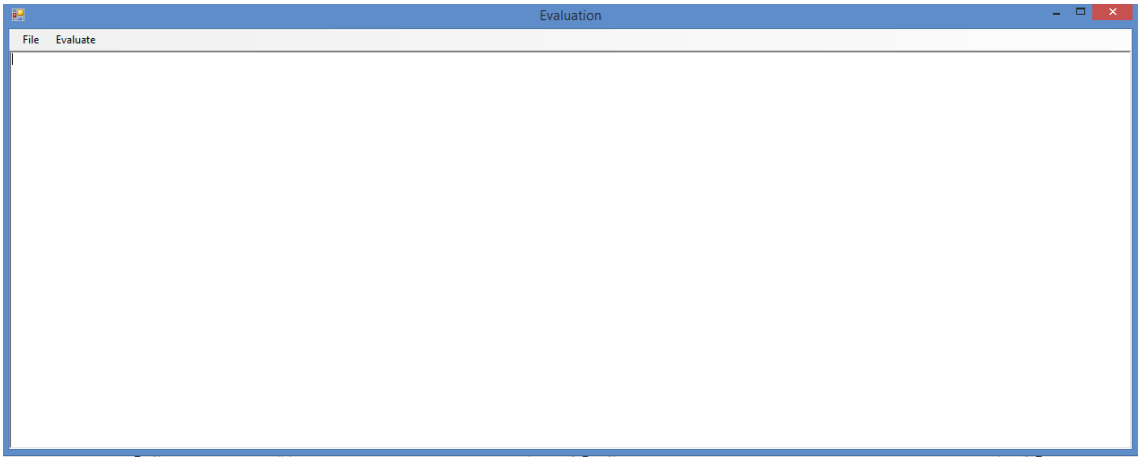

Figura 27 Ventana de Evaluation (evaluador del clasificador de opiniones).

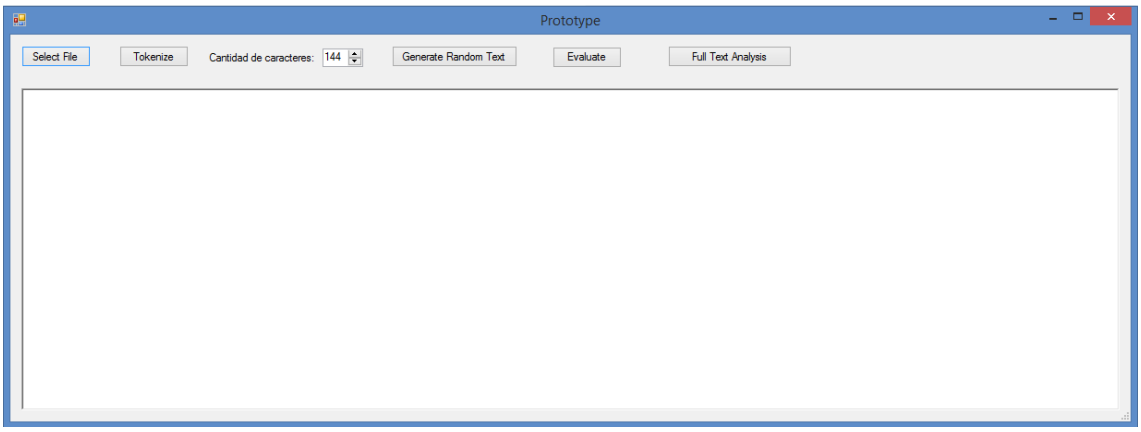

Figura 28 Ventana de MarcovChainTextGeneratorW.

El Launcher funciona como una ventana de bienvenida, la cual inicia el evaluador del clasificador de opiniones, que es la principal funcionalidad de la aplicación, y a modo de añadidura el generador de textos aleatorios con fines de clasificación.

El evaluador del clasificador sirve para seleccionar un *ground truth* en formato *blogging* o de *microblogging* previamente clasificado por humanos, y comparar el comportamiento del algoritmo de clasificación con respecto al *ground truth*.

El generador de texto aleatorio con cadenas de Markov, permite seleccionar un archivo de texto en lengua inglesa para generar la cadena de Markov, y luego generar textos aleatorios con el fin de evaluarlos con el algoritmo de clasificación. También permite realizar evaluaciones manuales en inglés del algoritmo TBONTB.

3.5.3.2. Gráfico de distribución de valencias medias

El gráfico de valencias medias es un formulario de windows con un control chart Windows [51] empotrado, el cual posee las siguientes series:

- Opiniones Positivas: Está compuesta por las valencias medias de las opiniones positivas. Se representa por medio de puntos anaranjados.
- Media Positiva: Es la valencia media positiva de las opiniones positivas. Se representa por medio de una línea horizontal de color anaranjado oscuro.
- Opiniones Negativas: Está compuesta por las valencias medias de las opiniones negativas. Se representa por medio de puntos verdes.
- Media Negativa: Es la valencia media negativa de las opiniones negativas. Se representa por medio de una línea horizontal de color azul oscuro.
- Umbral de decisión: Es el valor intermedio entre la valencia media positiva y negativa. Se representa por medio de una línea horizontal de color rojo.

Además de mostrar las series anteriormente descritas, para la evaluación del clasificador de opiniones usando los conjuntos de entrenamiento en formato *microblogging* y *blogging*, se añadieron varias funcionalidades adicionales, algunas de las cuales no son nativas del control, sino que se implementaron como métodos de extensión [52] de la funcionalidad del control tales como el zoom sobre una región rectangular del *chart area* [53], y el arrastrar el gráfico dentro del área de visibilidad del control; esta última funcionalidad en inglés se conoce como

panning. Estas funcionalidades por medio de métodos de extensión, se añadieron con una dll que me descargué del artículo [54] con licencia MIT [55].

Estas funciones están disponibles desde un menú contextual sobre el área del gráfico, como se muestra en la siguiente imagen:

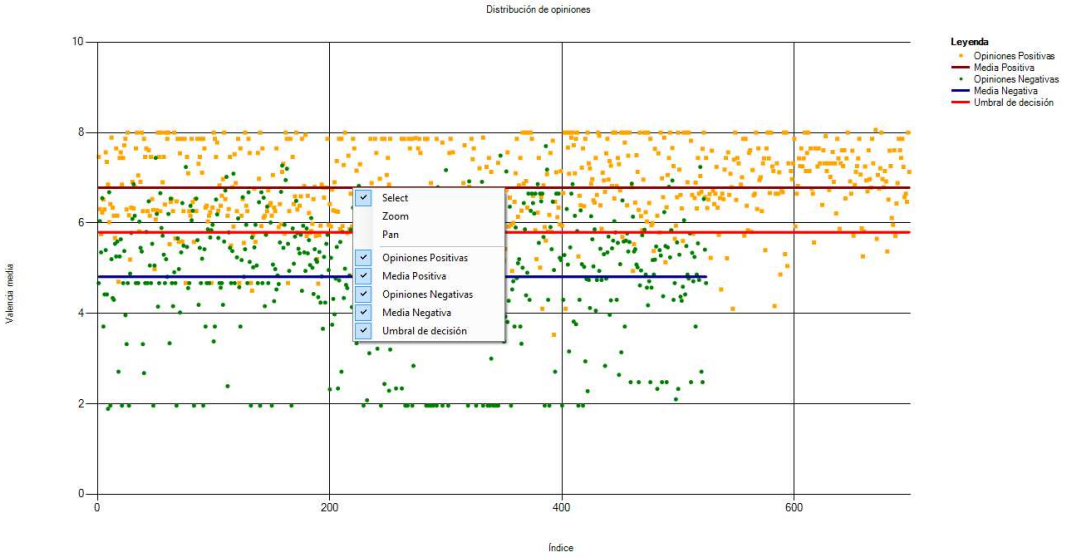

Figura 29 Menú contextual de la gráfica de distribución de las opiniones.

Desmarcando los nombres de las series, las mismas van desapareciendo del gráfico como se muestra en este caso con la serie “Opiniones Positivas”:

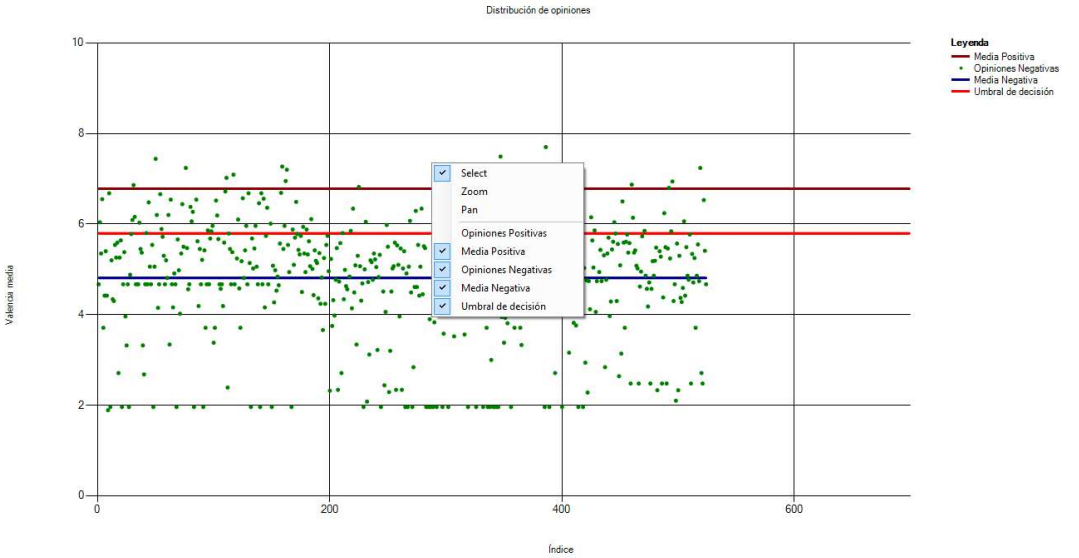

Figura 30 Gráfico de distribución de opiniones sin "Opiniones Positivas".

Para hacer zoom se marca la opción del menú “Zoom” y se selecciona el rectángulo a amplificar como se ve a continuación:

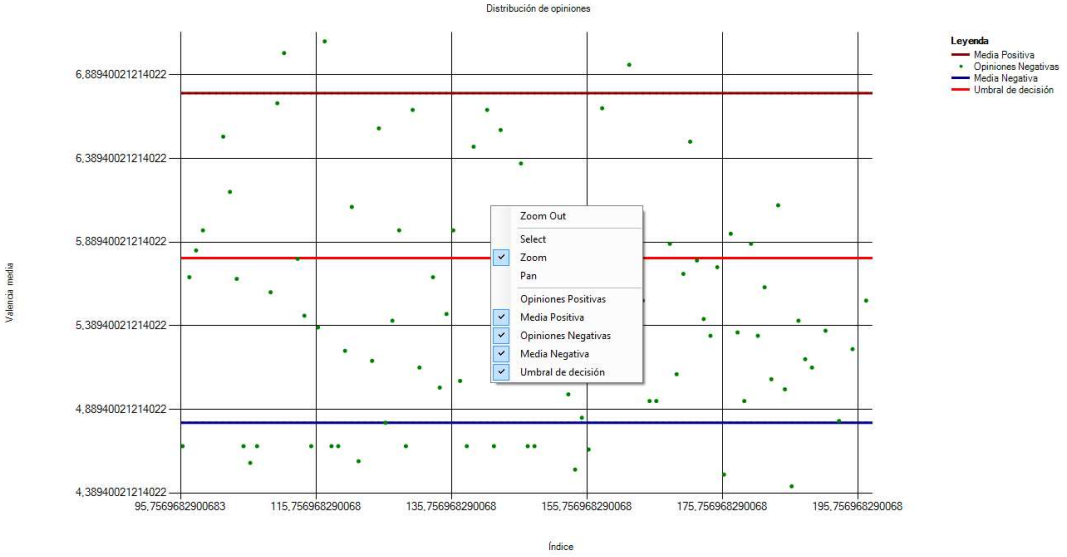

Figura 31 Zoom realizado en el gráfico de distribución de opiniones.

Una funcionalidad más que implementé fue la de mostrar el texto de la opinión como un *tooltip* cuando el mouse se detiene durante unos segundos sobre un punto correspondiente a la serie de "Opiniones Positivas" u "Opiniones Negativas", como se muestra a continuación:

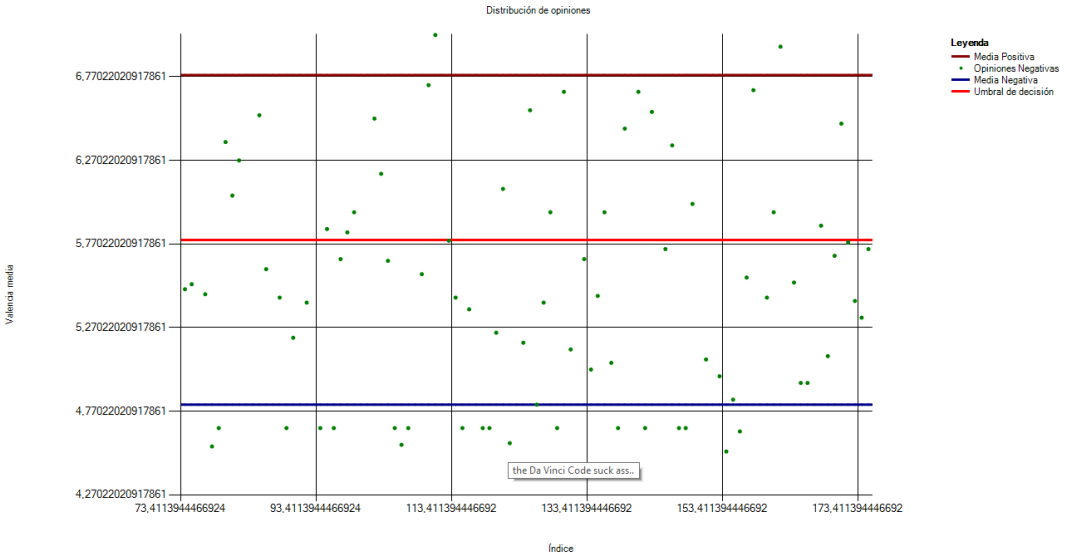

Figura 32 Tooltip que muestra el texto de la opinión positiva o negativa en el gráfico de distribución.

Esto lo implementé mediante el evento *GetTooltipText* [56] y haciendo uso de la propiedad *Tag* de cada *DataPoint* [57] para almacenar el texto de la opinión.

3.6. Implantación

3.6.1. Base de datos

Para implantar la base de datos, es necesario seguir los siguientes pasos:

- Crear una copia de seguridad [58] de la base de datos feelings tal y como se muestra en la siguiente imagen:

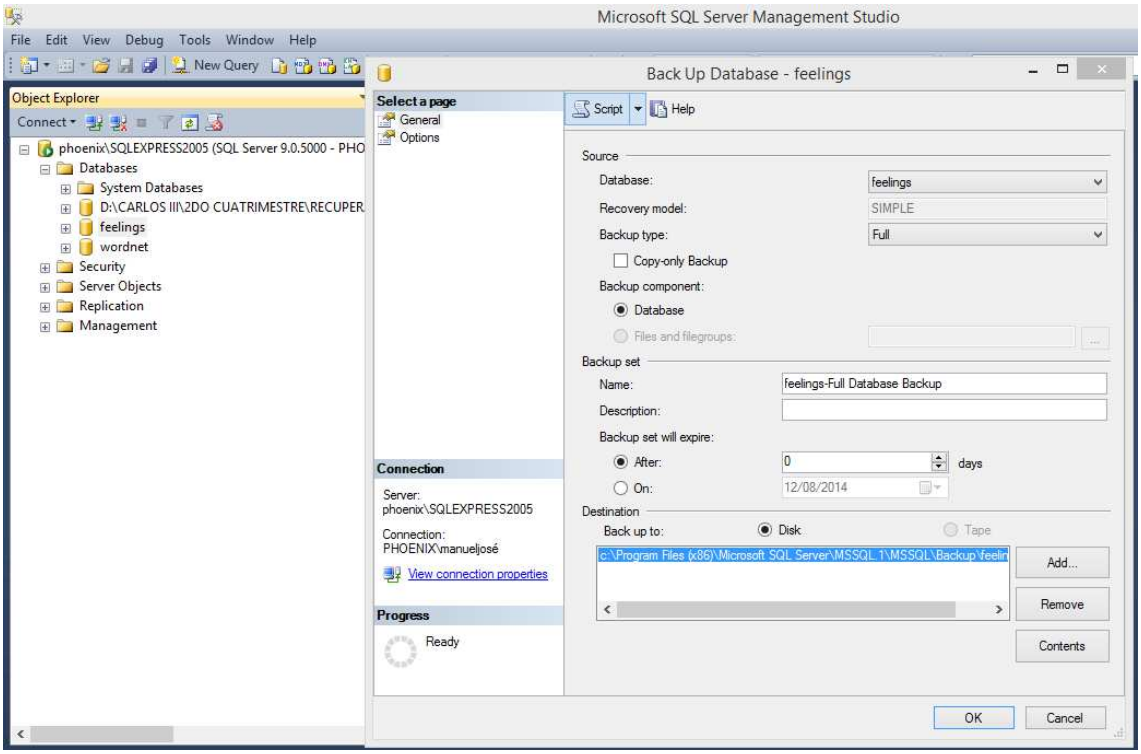

Figura 33 Copia de seguridad de la base de datos con Microsoft SQL Server Management Studio.

- Una vez creado el archivo feelings.bak, restaurar [59] la copia de seguridad en el SQL Server donde será instalada la aplicación desarrollada.
- Teclear en todos los App.config en cada una de las componentes de la aplicación la cadena de conexión a la base de datos feelings [60]. En el entorno local se ve así:

Figura 34 App.config en mi entorno local.

3.6.2. Publicación del código de la solución del proyecto

Dado que estamos ante una aplicación de escritorio basada en formularios de windows, se deben seguir los siguientes pasos para publicar [61]:

- Configurar la solución en modo *Release* (bandera de precompilación), tal y como se muestra en la imagen (deseleccionando *Debug* a *Release* en el combo):

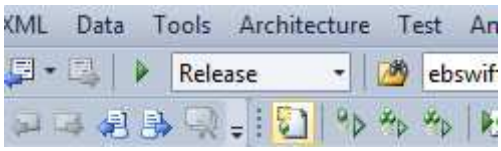

Figura 35 Modo release de la publicación.

- Se abre el diálogo de propiedades del proyecto Launcher como se muestra:

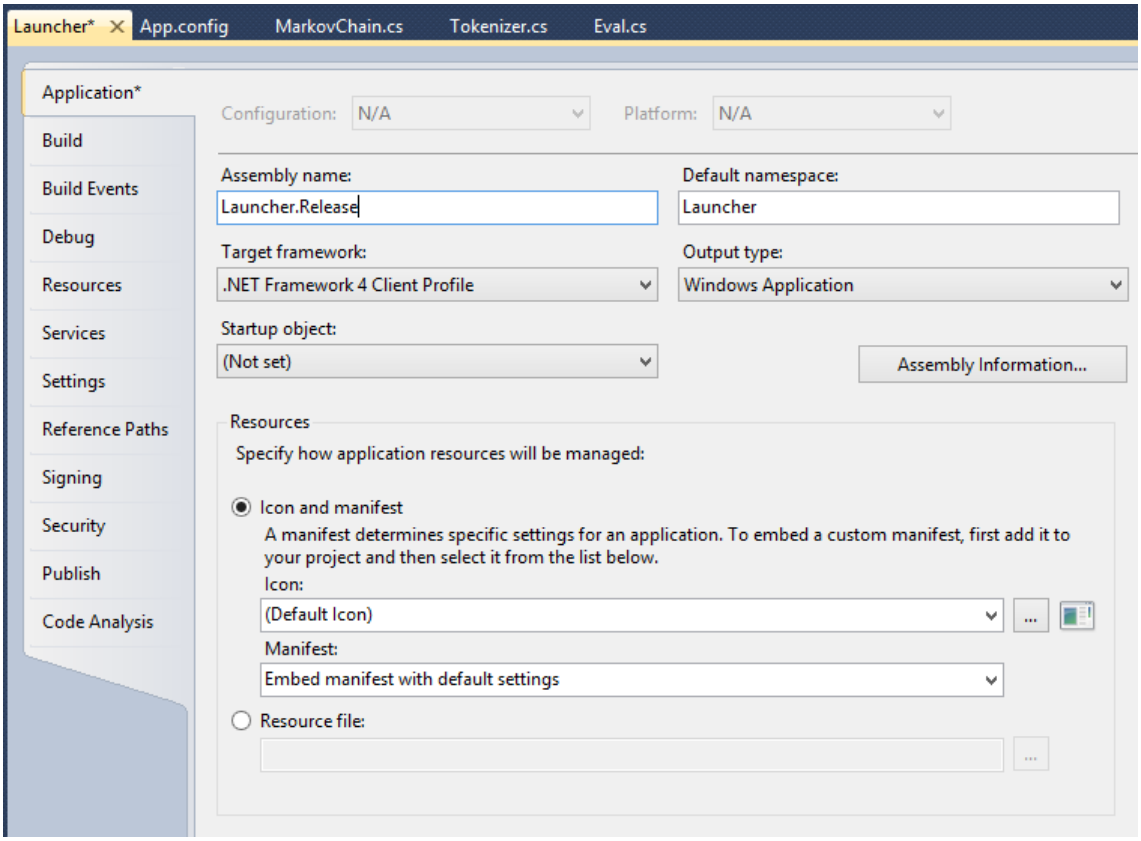

Figura 36 Propiedades del proyecto Launcher, que es desde donde se publica.

Obsérvese que el nombre del ensamblado incluye el *flag* (bandera) de precompilación *Release* después del nombre del proyecto Launcher.

- Ve a la opción en el menú vertical izquierdo *Publish,* tal y como se muestra a continuación:

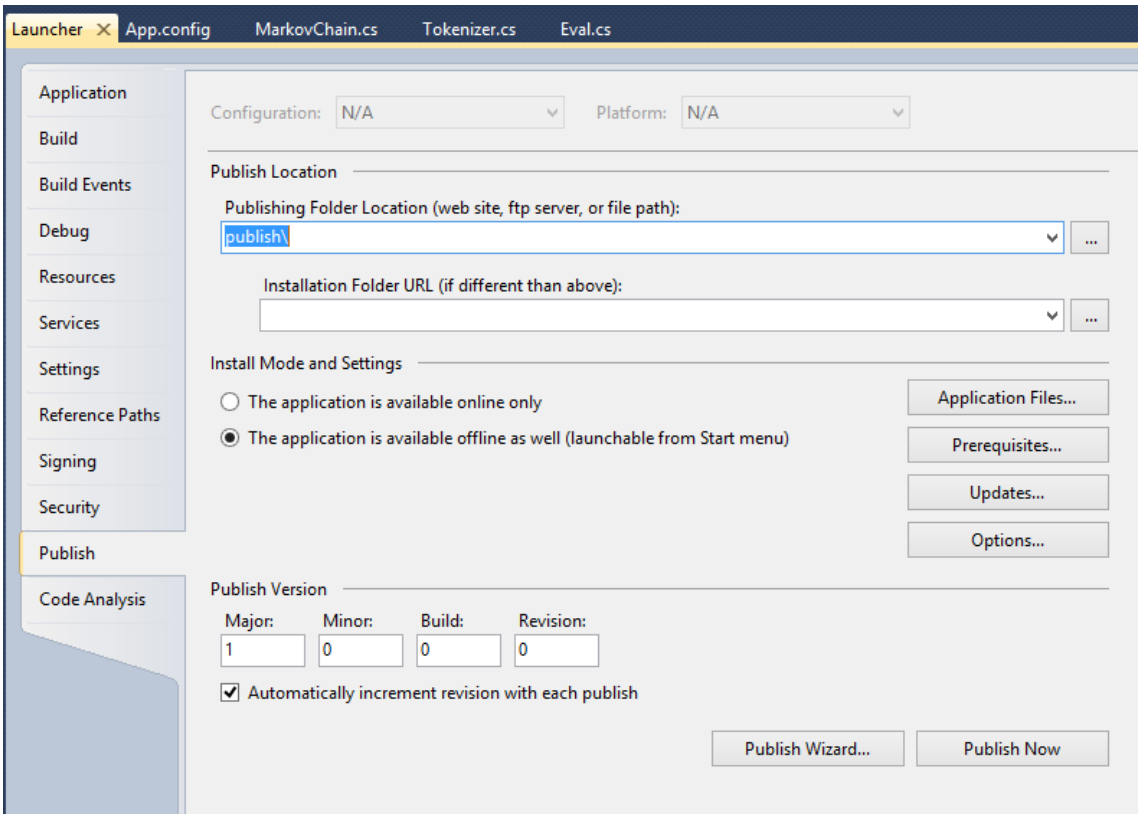

Figura 37 Pantalla de publicación de la aplicación en las propiedades del proyecto Launcher.

Nótese los siguientes detalles de la publicación:

- Ruta de publicación donde se va a generar el instalador.
- Ruta de instalación desde donde se descarga el instalador en el *browser* y posteriormente se instala, o simplemente la carpeta donde se guarda el *setup* de la aplicación.
- Número de versión de la aplicación. Dado que es un prototipo, sería la versión 1.0.0.0.

3.6.3. Licencia del Proyecto

La licencia que escogí es la del MIT [62], la cual permite reutilizar el software de mi proyecto como software libre o como software no libre, y es compatible con cualquier otra licencia para las modificaciones futuras que se quieran realizar sobre el mismo.

A continuación una imagen del encabezado de la licencia en cada archivo de código fuente desarrollado por mi:

```
//The MIT License (MIT)

//Copyright (c) <2014> <Manuel José Lazo Reyes>

//Permission is hereby granted, free of charge, to any person obtaining a copy
//of this software and associated documentation files (the "Software"), to deal
//in the Software without restriction, including without limitation the rights
//to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
//copies of the Software, and to permit persons to whom the Software is
//furnished to do so, subject to the following conditions:

//The above copyright notice and this permission notice shall be included in
//all copies or substantial portions of the Software.

//THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
//IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
//FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
//AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
//LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
//OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
//THE SOFTWARE.
```

Figura 38 Encabezado de licencia MIT de mi proyecto.

3.6.4. *URL* de descarga del backup de la base de datos "*feelings*"

La *url* de descarga del *backup* de 165 Mb de la base de datos del proyecto se encuentra en mi *google drive* de manera pública (no requiere autentificación) y es:
https://drive.google.com/open?id=0B33oo9gsYX5fLTE1M3ZYODYtOHM&authuser=0 .

3.6.5. *URL* de descarga del conjunto de entrenamiento en formato *microblogging*

Igualmente se encuentra en mi *google drive* (no requiere autentificación) y es:
https://drive.google.com/open?id=0B33oo9gsYX5fN3ZWc0x2eXM0a2M&authuser=0 .

3.6.6. *URL* de descarga del conjunto de entrenamiento en formato *blogging*

También está disponible en mi *google drive* (sin autentificación) y es:
https://drive.google.com/open?id=0B33oo9gsYX5feFI1bkVLdEgwZGc&authuser=0 .

3.6.7. *URL* de descarga del código fuente de la aplicación desarrollada

Para mantener un control de versionado del código fuente, en futuras modificaciones que se le vayan añadiendo al proyecto, decidí crearme un repositorio privado en assembla [63], con control de código fuente en subversión. Como software de control de código fuente, recomiendo TortoiseSVN, el cual es un cliente Apache ™ Subversion (SVN) ®, implementado como una extensión de *shell* para el sistema operativo *Windows*. Es intuitivo y muy fácil de usar, ya que no requiere del cliente de línea de comando de Subversion para funcionar. Y es libre de utilizar, incluso en un entorno comercial.

Apache Subversion (a menudo abreviado SVN) es un sistema de control de versiones de software distribuido como software libre bajo la licencia Apache. Los desarrolladores usan

Subversion para mantener las versiones actuales e históricas de los archivos, como código fuente, web páginas, y la documentación.

Los pasos a seguir para obtener el código fuente son:

- Paso 1: Descargar *tortoisesvn* [64].
- Paso 2: Instalar *tortoise*.
- Paso 3: Exportar el código fuente del proyecto desde la url (click derecho sobre la carpeta de *windows* donde desea realizar la exportación y elegir la opción "*Export*"):

https://subversion.assembla.com/svn/tbontb/ . A continuación una imagen:

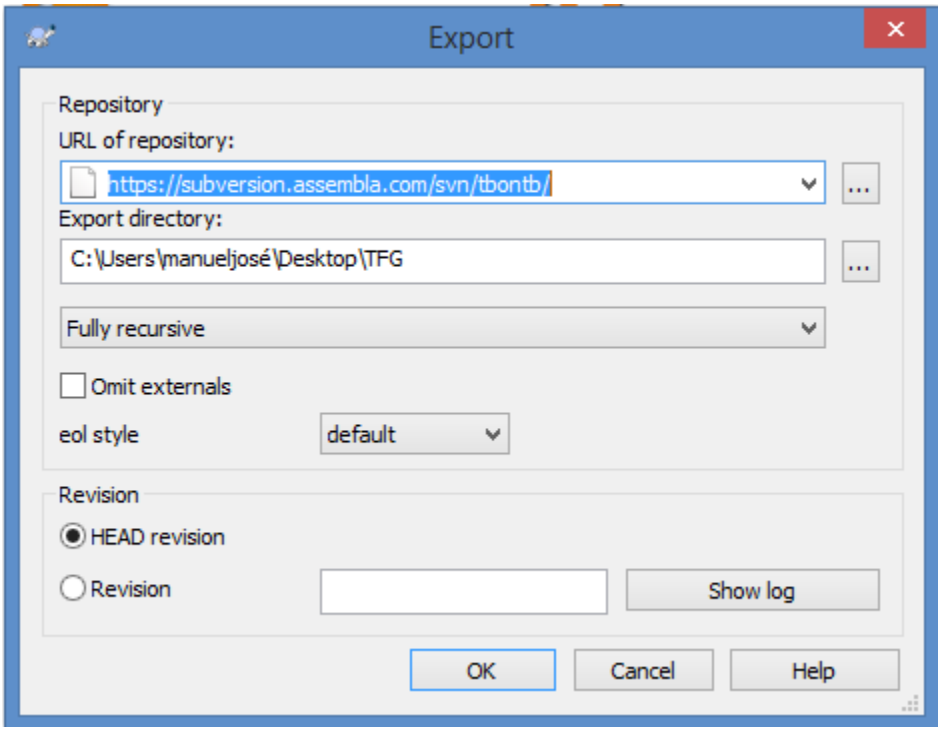

Figura 39 Obtención de código fuente con tortoisesvn.

Una vez exportado el código ya está todo hecho. Simplemente es necesario ejecutar el visual studio, para cargar la solución del trabajo final.

4. Evaluación

En este apartado se van a especificar las pruebas a realizar para comprobar que la aplicación funciona correctamente y cumple todas las funcionalidades descritas anteriormente. Todas estas pruebas se han ido realizando iteración a iteración, pero una vez terminada la aplicación se va a comprobar que se siguen pasando todas las pruebas.

4.1. Representación de la evaluación

La evaluación del clasificador se representa en una estructura de datos tipo diccionario del tipo `Dictionary<string, Tuple<int, int?, double?, bool>>`. A continuación una representación gráfica de lo que significa:

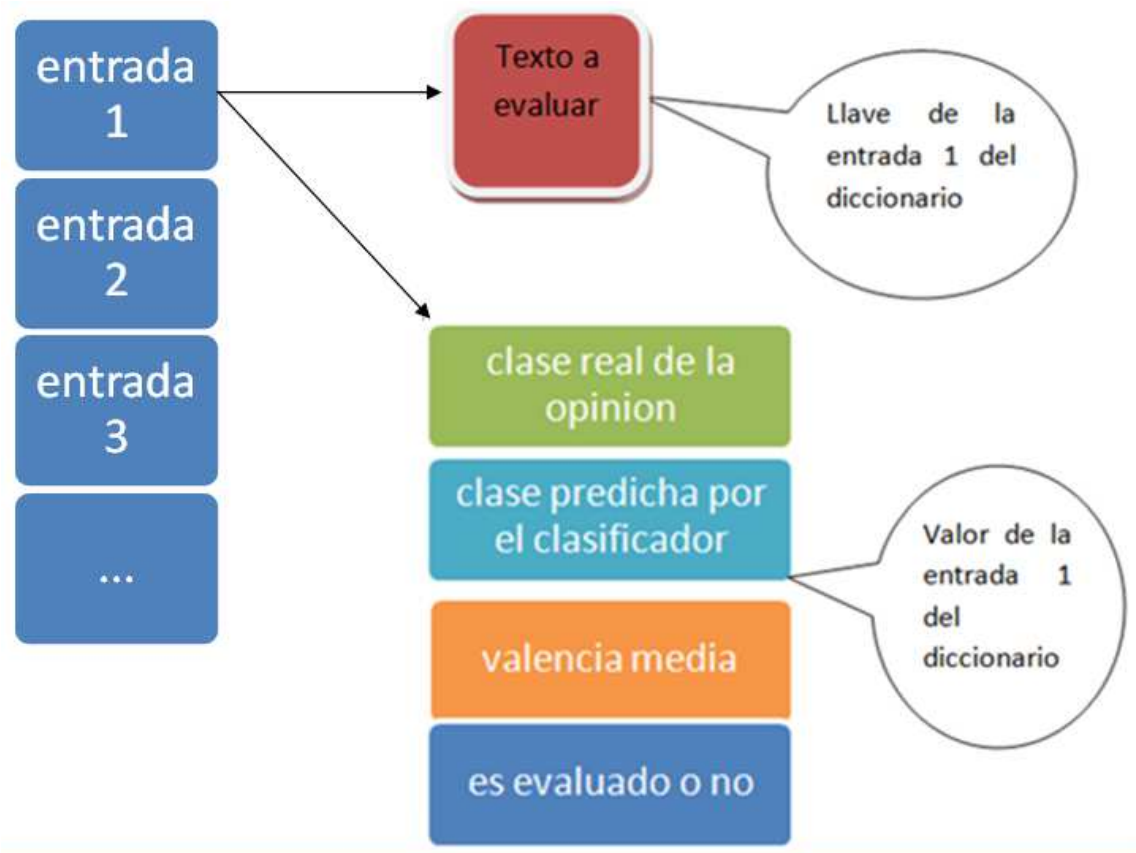

Figura 40 Representación del diccionario de evaluación de opiniones.

Es necesario aclarar algunas cuestiones acerca de esta representación:

- La clase real de la opinión es la que está asociada al texto correspondiente del archivo de entrenamiento. Solo puede ser “positiva” o “negativa”.
- La clase predicha por el clasificador es la que devuelve el clasificador. Puede ser “positiva” o “negativa” o vacía en caso de que el clasificador no encuentre coincidencias entre la lista de palabras afectivas y las que conforman el texto de la opinión.

- La valencia media es un número real entre 1 y 9, que es el promedio de todas las palabras afectivas encontradas por el clasificador en el texto de la opinión. Puede ser nula o vacía, en caso de que no existan palabras afectivas en el texto de la opinión.
- Aparecerá *true,* en caso de que se encuentren palabras afectivas en el texto de la opinión, o *false* en caso contrario.

Resumiendo, toda la evaluación que se hace sobre un conjunto de instancias de textos a evaluar, se almacena en un diccionario, donde cada entrada al mismo es una estructura llave – valor, en la cual la llave contiene el texto a evaluar y el valor es una tupla ordenada que posee en el siguiente orden la clase real del texto, la clase predicha, la valencia media del texto si ha sido clasificado por el clasificador y finalmente un *flag* booleano que indica si ha sido clasificado el texto.

4.1.1. Medidas de evaluación

Las medidas de evaluación obtenidas son:

- Porcentaje de acierto: Se trata de la razón entre la cantidad de coincidencias entre la clase real y la clase predicha, con la cantidad de opiniones clasificadas por el clasificador.
- Porcentaje de acierto positivo: Es la razón entre la cantidad de coincidencias entre la clase real de instancias de textos “positivos” y la clase predicha, con la cantidad de opiniones clasificadas por el clasificador como “positivas”.
- Porcentaje de acierto negativo: Es la razón entre la cantidad de coincidencias entre la clase real de instancias de textos “negativos” y la clase predicha, con la cantidad de opiniones clasificadas por el clasificador como “negativas”.
- Porcentaje de textos no clasificados: Es la razón entre la cantidad de opiniones no clasificadas por el clasificador y el total de textos en el conjunto de entrenamiento.
- Cantidad de positivos: Se refiere a la cantidad de instancias clasificadas cuya clase real es “positiva”.
- Cantidad de negativos: Se refiere a la cantidad de instancias clasificadas cuya clase real es “negativa”.
- Valencia media global: No es más que el promedio de la valencia media de todas las opiniones clasificadas por el clasificador.
- Desviación media global: Es la desviación media de la valencia media de todas las opiniones clasificadas por el clasificador.
- Valencia media positiva global: Trata del promedio de la valencia media de todas las opiniones clasificadas por el clasificador, cuya clase real es “positiva”.
- Desviación estándar positiva: Es la desviación estándar de la valencia media positiva de todas las opiniones clasificadas por el clasificador.
- Valencia media negativa global: Trata del promedio de la valencia media de todas las opiniones clasificadas por el clasificador, cuya clase real es “negativa”.
- Desviación estándar negativa: Es la desviación estándar de la valencia media negativa de todas las opiniones clasificadas por el clasificador.

4.2. Evaluación de opiniones en formato microblogging

Para evaluar el clasificador de opiniones en formato *microblogging* me descargué de la competición [62] de *Information Retrieval* organizada por la universidad de Michigan, el conjunto de entrenamiento de la competición, puesto que vienen previamente clasificadas.

Cada opinión del conjunto de dicho entrenamiento es una frase extraída de medios sociales (blogs). Los datos de entrenamiento contienen 7.086 frases, ya etiquetados con 1 (sentimiento positivo) o 0 (sentimiento negativo). Los datos de prueba contienen 33.052 frases que están desprovistos de la clase de polaridad emocional, razón por la cual no nos interesan para la evaluación. Nos enfocaremos entonces con el conjunto de datos de entrenamiento de la competición, porque es el que nos vale para evaluar cuan bueno es el clasificador de opiniones.

A continuación se muestra un fragmento del conjunto de entrenamiento de opiniones positivas:

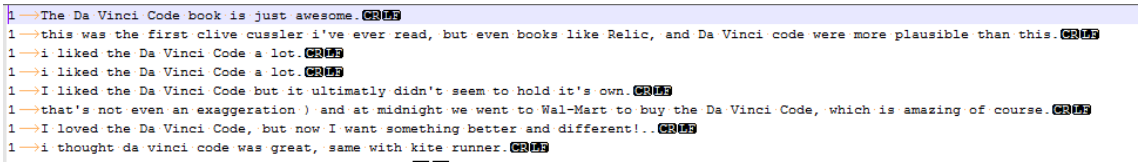

Figura 41 Fragmento de opiniones positivas del conjunto de entrenamiento en formato microblogging.

Como se puede apreciar esta colección presenta ruido porque contiene frases repetidas. Las repetidas se evalúan una sola vez. Otro factor a tener en cuenta es que para algunas frases no se reconoce ninguna palabra de la lista de palabras afectivas almacenadas en la base de datos, por lo que no es posible clasificar dicha frase.

A continuación también se muestra un fragmento del conjunto de entrenamiento de opiniones negativas:

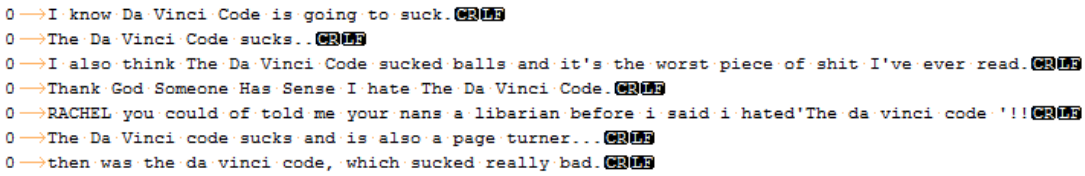

Figura 42 Fragmento de opiniones negativas del conjunto de entrenamiento en formato microblogging.

4.2.1. Resultados

Los resultados de la evaluación del conjunto de entrenamiento en formato *microblogging* han superado todas mis expectativas. A continuación se listan las medidas de evaluación:

- El porcentaje de acierto es 84%.
- El porcentaje de acierto positivo es 87%.
- El porcentaje de acierto negativo es 79%.
- El porcentaje de textos no clasificados es 15%.
- La cantidad de textos positivos es 699.

- La cantidad de textos negativos es 524.
- La valencia media global entre 1 y 9 es 5,94.
- La desviación estándar global es 1,48.
- La valencia media positiva entre 1 y 9 es 6,78.
- La desviación estándar positiva es 0,88.
- La valencia media negativa entre 1 y 9 es 4,81.
- La deviación estándar negativa es 1,36.

Estos resultados se pueden apreciar en la siguiente imagen:

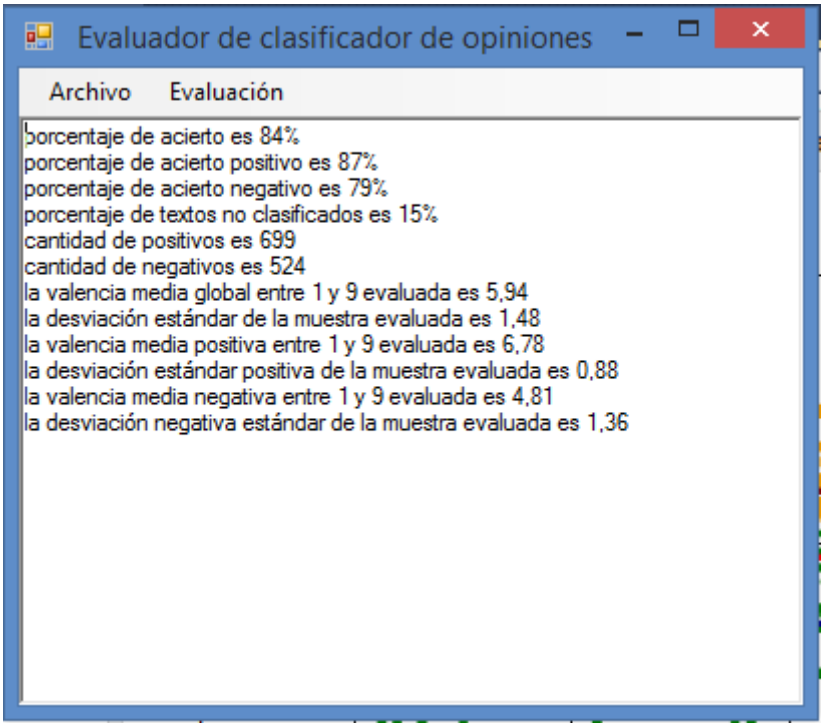

Figura 43 Resultados de la evaluación del conjunto de entrenamiento de opiniones en formato microblogging.

En la siguiente gráfica, se muestran estos resultados para todas las opiniones evaluadas en el formato *microblogging*:

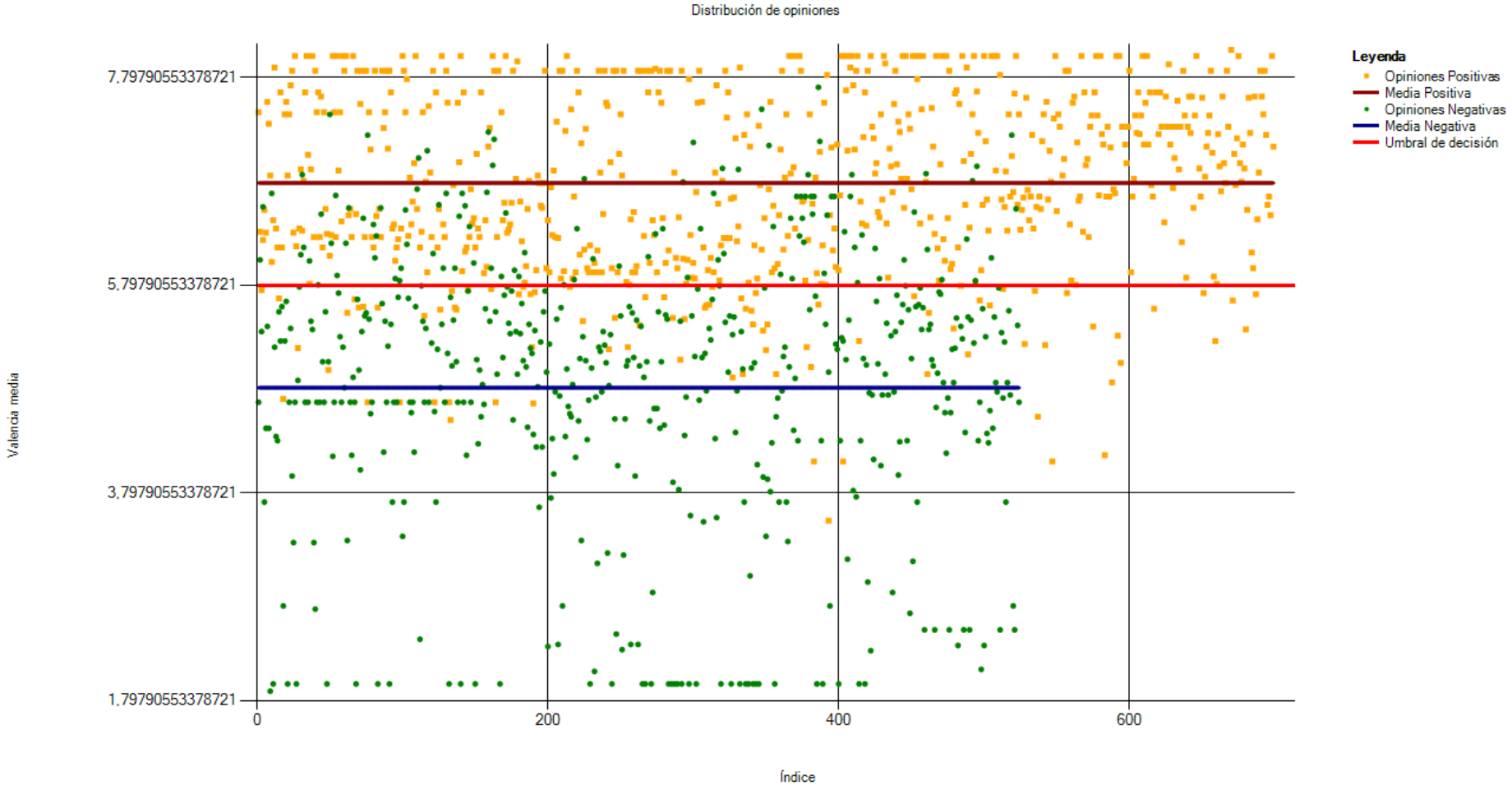

Figura 44 Gráfica de evaluación del clasificador con el conjunto de entrenamiento en formato microblogging.

Como se puede apreciar en la leyenda de la gráfica, los puntos anaranjados son opiniones cuya clase predicha es “positiva”, mientras que los puntos verdes son opiniones cuya clase predicha es “negativa”. Cabe señalar, que en esta gráfica no está representada el 15% de los textos en formato *microblogging* del conjunto de entrenamiendo, debido a que no contienen palabras de la lista en base de datos de palabras afectivas. Tampoco están los textos repetidos, ya que solo se clasifican una vez.

La línea de color anaranjado oscuro, es la valencia media de las opiniones que tienen una clasificación dada por el clasificador, pero cuya clase real es “positiva”. Como se puede ver el valor indicado es 6,78 aproximadamente. En cambio la línea de color azul oscuro es la valencia media de las opiniones que también son clasificadas, pero en cambio su clase real es “negativa” con un valor de 4,81. Finalmente, la línea roja es el umbral de decisión de la polaridad emocional de la opinión en cuestión. El umbral se corresponde con el valor intermedio entre las valencias medias positiva y negativa respectivamente. Para ver un video demostrativo pinchar en el link https://www.youtube.com/watch?v=RhkrFl3oGoI&list=UUYNmcO-KAHchzJxI7hup0lw .

4.3. Evaluación de opiniones en formato blogging

De una web de datos de crítica cinematográfica [63] me descargué un *dataset* ya clasificado de críticas de cine que ya están clasificadas en “positiva” o “negativa” y con la característica añadida de que el texto a evaluar contiene muchas más palabras. Estamos hablando de un párrafo de muchas oraciones escritos además en un vocabulario propio del lenguaje cinematográfico, el cual es bastante específico y tiene su propia semántica.

Cabe señalar que fue necesario crear un módulo aparte, solo para unificar todas las críticas en un solo archivo con el mismo formato anterior de 1 o 0, luego tabular con un carácter de tabulación y finalmente el texto, ya que el comprimido una vez descompactado crea dos directorios, uno pos para las “positivas” y otro neg para las “negativas” respectivamente.

A continuación un fragmento de opiniones del conjunto de entrenamiento que son positivas:

1⟶the others (2001) nicole kidman , christopher eccleston , fionnula flanagan , elaine cassidy , eric sykes , alakina mann , james bentley , rene ascherson . written and directed by alejandro amen□bar . 104 minutes . rated pg-13 , 3 stars review by ed johnson-ott , nuvo newsweekly www . nuvo . net archive reviews at http : //reviews . imdb . com/reviewsby ? edward + johnson-ott to receive reviews by e-mail at no charge , send subscription requests to ejohnsonott@prodigy . net or e-mail ejohnsonott-subscribe@onelist . com with the word " subscribe " in the subject line . it's hard not to recommend " the others . " the supernatural thriller , written and directed by alejandro amen□bar (" open your eyes ") , adroitly establishes and maintains a low-key atmosphere of menace . the cinematography , by javier aguirresarobe , is as good as i have ever seen and nicole kidman gives another in her growing body of fine performances . but the pacing of the story moves from deliberate to downright sluggish and the payoff of the tale left me less than satisfied . overall , the film plays like a very high quality version of any number of old " twilight zone " episodes where the characters spend a great deal of time wandering about looking afraid and disoriented , only to learn they are actually a child's toy , a military test subject or a department store dummy . watching those vintage shows and listening to the players chatter , i always wanted to shout , " get on with it ! " as much as i appreciated the atmosphere and acting in " the others , " my reaction was much the same . set at an island mansion off the coast of england during world war ii , the story focuses on grace (kidman) , who tends to her children anne (alakina mann) and nicholas (james bentley) and worries about her husband , charles (christopher eccleston) , a missing serviceman . anne and nicholas suffer from photosensitivity and grace patrols the estate with the keys to all 50 doors , protecting the little ones from excess light by making sure that only one door is open at a time . at the beginning of the film , three servants , mrs . mills (fionnula flanagan) , young , mute lydia (elaine cassidy) , and mr . tuttle (eric sykes) , an aging gardener , join the family . the two that speak seem agreeable enough at first , but it soon becomes apparent that they know something that grace does not . to make matters worse , the children are upset : nicholas is unusually jittery and anne claims to be seeing ghosts . grace attempts to blame the troubles on the new arrivals to her home , only to realize that whatever is happening is beyond them . that's essentially the whole story , with the tension growing until the pivotal moment when everything becomes clear . earlier , there is an outstanding scene where grace darts outside , only to be enveloped in fold after fold of shimmering fog . the visuals in the otherworldly sequence , courtesy of aguirresarobe , are simply astounding . i also enjoyed the presence of religion , a rarity in films dealing with the supernatural . grace is a christian and answers her children's questions about life and death with the assurance of a devout worshiper . when mother is away , though , the kids speculate whether her statements are fact or folklore , just as real children do . but those nice touches fail to enliven a film that is too slow or make up for a lackluster ending . " the others " sets out to be a classic ghost story , but fails to grasp that special something that makes such films more than layers of mist . CRLF
1⟶my fellow americans is a movie that at first glance looks to have little substance (or a movie that we've all seen a million times) , two lifetime rivals thrown together and then the fun begins . this is exactly what happened in this movie , but fortunately , they managed to do it in an interesting and funny way . the movie starts with a quick (and i do mean quick) glance of two presidents russell kramer (jack lemmon) and matt douglas (james garner) . william haney (dan aykroyd) and ted matthews (john heard) are the new president and vice president . there is a scandal that arises involving a kickback from a contractor and haney is positive that he buried that years ago . he finds a scapegoat in kramer and now everyone wants kramer and douglas dead . this movie was exceptional for many reasons . one being that they found people (lemmon and garner) that have good chemisty together . they worked very well as a unit and they mirrored each other perfectly , one being a ladies man and one being the old man (i'll let you figure which is which) . also , they found people that know their parts as govenment officials well . it seemed to me that garner played almost exactly the same role that he played in the distinguished gentleman (except then he was a congressman) . experience counts for a lot ! ! CRLF
1⟶mute witness (1994) a film review by justin felix . copyright 1999 justin felix . any comments about this review ? contact me at justinfelix@yahoo . com all of my film reviews are archived at http : //us . imdb . com/m/reviews_by ? justin + felix written and directed by anthony waller . starring marina zudina . special cameo by alec guinness . rated r (contains brutal violence , nudity , and profanity) 90 mins . synopsis : an attractive mute makeup artist , working on an ultra-cheesy slasher movie in moscow , witnesses the production of a brutal snuff film and is subsequently chased by really bad russians . meanwhile , the artist's sister and boyfriend clumsily try to save her . comments : mute witness came as a surprise to me the first time i watched it . drawn by the clever artwork on the video box , i rented the film expecting a complete turkey . mute witness , however , was original , offbeat , and well-made . it's one of those cool little finds that no one seems to know about . i've subsequently found it at most video rental places i visit , and it may be seen , on occasion , on the independent film channel . the first hour of mute witness is extremely tense , as billy , the quite believable mute heroine , sees members of the

Figura 45 Fragmento de opiniones positivas en formato blogging.

Como se aprecia no contienen saltos de líneas y cada párrafo posee varias oraciones largas, por lo general.

En la siguiente imagen se muestra también un fragmento pero de opiniones negativas:

0⟶director : luis llosa cast : jennifer lopez , ice cube , owen wilson , eric stoltz humanities quest for knowledge never ends . so a team of scientists and film-makers travel to the amazon to search for a legendary indian tribe . the party consists of anthropologist steven cale (eric stoltz) and the camera team consisting of terri flores (jennifer lopez) , danny rich (ice cube) , gary dixon (owen wilson) , denise kahlberg (kari wuhrer) and warren westridge (jonathan hyde) . early on their journey they meet paul sarone (jon voight) whose boat is stuck on the shore . they agree to give him a ride to the next village . he claims to know the area well and can be useful locating the native tribe . very soon their friendliness backfires on the group because sarone turns out to be a snake hunter without scruples who only wants to catch a giant anaconda and sell it to a zoo . we don't have to wait too long for the giant snake . she just had a panther hors d'oevre and now is looking for the main course . our heroes paddle around in the amazonas as if it were the pool in their own backyard . no wonder giant animals mistake their splashing for a dinner bell . our anaconda is a polite one and swallows the first victim in one big gulp . enjoy ! so much for the first attempt to catch her . but who would want to catch a giant snake with a fishing pole ? our villain sarone shows his soft side when he stops terri from shooting the snake . too bad that anaconda is just about to strangle another member of the expedition . one by one she goes after the others . eric stoltz is stung by a giant wasp right in the beginning and is mercifully unconcious for the rest of the adventure . the rest of the crew keeps entertaining the viewer although not the way the makers of the movie had planned . however the scenes without the anaconda are rather boring . whenever the leading lady shows up we're in for a laugh . the snake reminds us of a favorite character of a famous animated movie even if she should be an awe-inspiring monster . her attacks always follow the same plan : one last hypnotic look - she's looking at you , kid - then she speedily wraps herself around her victim and starts to gush it down . mostly we don't see the act of devouring . but she looks nice when she wiggles away with her bulging middle part . whoever did the special effects on this movie may have wanted to go to a zoo first and study some real snakes . maybe then the anaconda model would have looked more real . the animatronics are somewhat more believable . but that didn't work for the strangling scenes . don't go see the movie for the f/x . they are everything but up-to-date . the viewer who likes to watch the end credits will see to his/her surprise that a snake expert was a consultant for the team . we may doubt though that he has ever seen the final result of his work . a well known american science magazine is also mentioned in the credits , but i will refrain from naming it here to avoid further damage to its reputation . the majority of viewer will have left the theater as soon as the credits start rolling , anyway . what kind of audience is the target group for this movie ? hard to say . this can't be a serious horror movie , or can it ? see for yourself . CRLF
0⟶species is a forehead-knockingly bad movie . you know what i'm talking about : that's when you sit there , slack-jawed , staring at the screen , and then just pound your first into your forehead in total disbelief at what you're witnessing . it's so awful it's downright charming ; it's about as bad and as * loud * as stargate . here's the " plot " : scientists recieve a transmission from space that seems to consist of a genetic code . after synthesizing it and crossbreeding it with human dna , they develop sil : an innocent-looking , wide-eyed girl-thing who , for reasons too protracted to list here , has to be destroyed . only trouble is , they can't simply put a gun to the back of her head and turn her into dog food -- like a bond movie , they have to kill her * elaborately * , which gives her a chance to escape . the government corralls together a team of " experts " -- mostly at screwing up , from the look of it : an assassin (michael madsen , looking even dopier than he did in resevoir dogs) , an empath (forest whitaker , looking like the pillsbury doughboy) , another scientist , and the creator of sil (ben kingsley , the best thing in the movie) . most of them will of course be murdered in various creative ways , but not before they get to show their various " skills " . smithson , the empath , for instance , has the uncanny ability to just spit out where sil is and what she's doing . thanks to the fact that his skill doesn't have any explicit rules , it leaves us scratching our heads as to why he doesn't just produce a map and draw an x on it . sil , as it turns out , is growing and developing at a furious rate -- which , of course , requires that she murder several people to feed herself . which is nothing compared to the vigor she exhibits when trying to find a mate . the adult sil (played by natasha henstridge) looks like she just walked off the runway of a fashion show and has no trouble finding nookie in california , and the movie gets some good laughs out of everyone reacting to her total guilelessness . there is also one moment -- maybe salvaged from an earlier , better draft of the script -- where she tries to to explain who she is , and can't . but the main purpose of the movie is never in doubt : violence , gratuitous nudity , ear-hammering sound effects . the last one is really ticking me off in a lot of movies lately : why , for instance , do we alwasy have swooshing sound effects to accompany the image of a flashlight playing across the camera lens ? light beams didn't disturb the air much the last time i checked . CRLF

Figura 46 Fragmento de opiniones negativas en formato blogging.

4.3.1. Resultados

Los resultados de la evaluación del conjunto de entrenamiento en formato *blogging*, no han sido tan espectaculares como los obtenidos para el formato *microblogging*, mas todo esto tiene su por qué, el cual se verá en el siguiente apartado. A continuación se listan las medidas de evaluación:

- El porcentaje de acierto es 59%.
- El porcentaje de acierto positivo es 60%.
- El porcentaje de acierto negativo es 59%.
- El porcentaje de textos no clasificados es 0%.
- La cantidad de textos positivos es 699.
- La cantidad de textos negativos es 698.
- La valencia media global entre 1 y 9 es 5,73.
- La desviación estándar global es 0,72.
- La valencia media positiva entre 1 y 9 es 5,77.
- La desviación estándar positiva es 0,21.
- La valencia media negativa entre 1 y 9 es 5,68.
- La deviación estándar negativa es 0,19.

Estos resultados se pueden apreciar en la siguiente imagen:

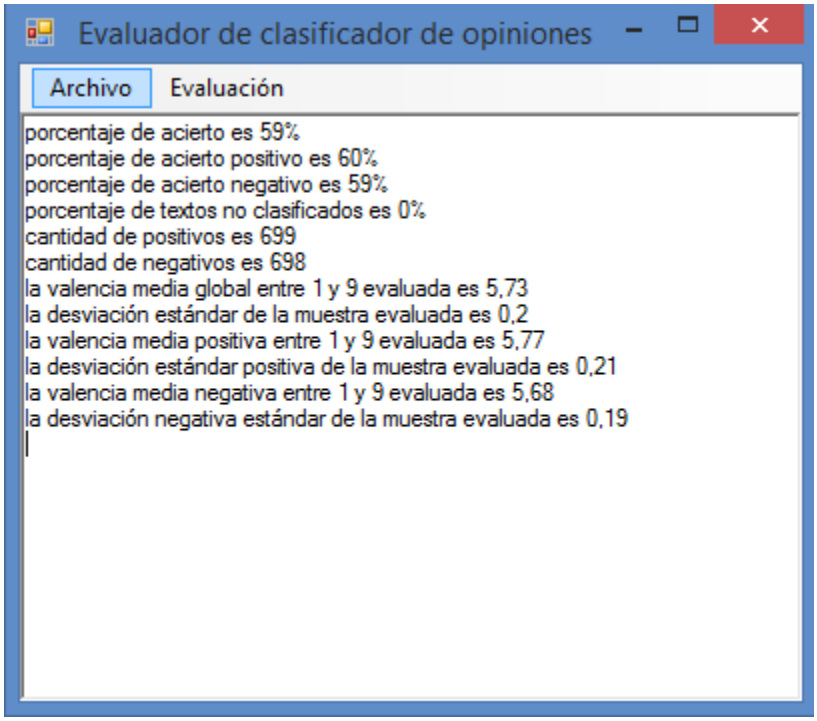

Figura 47 Resultados de la evaluación del conjunto de entrenamiento de opiniones en formato blogging.

En la siguiente gráfica se muestran los antes mencionados resultados obtenidos para el formato blogging:

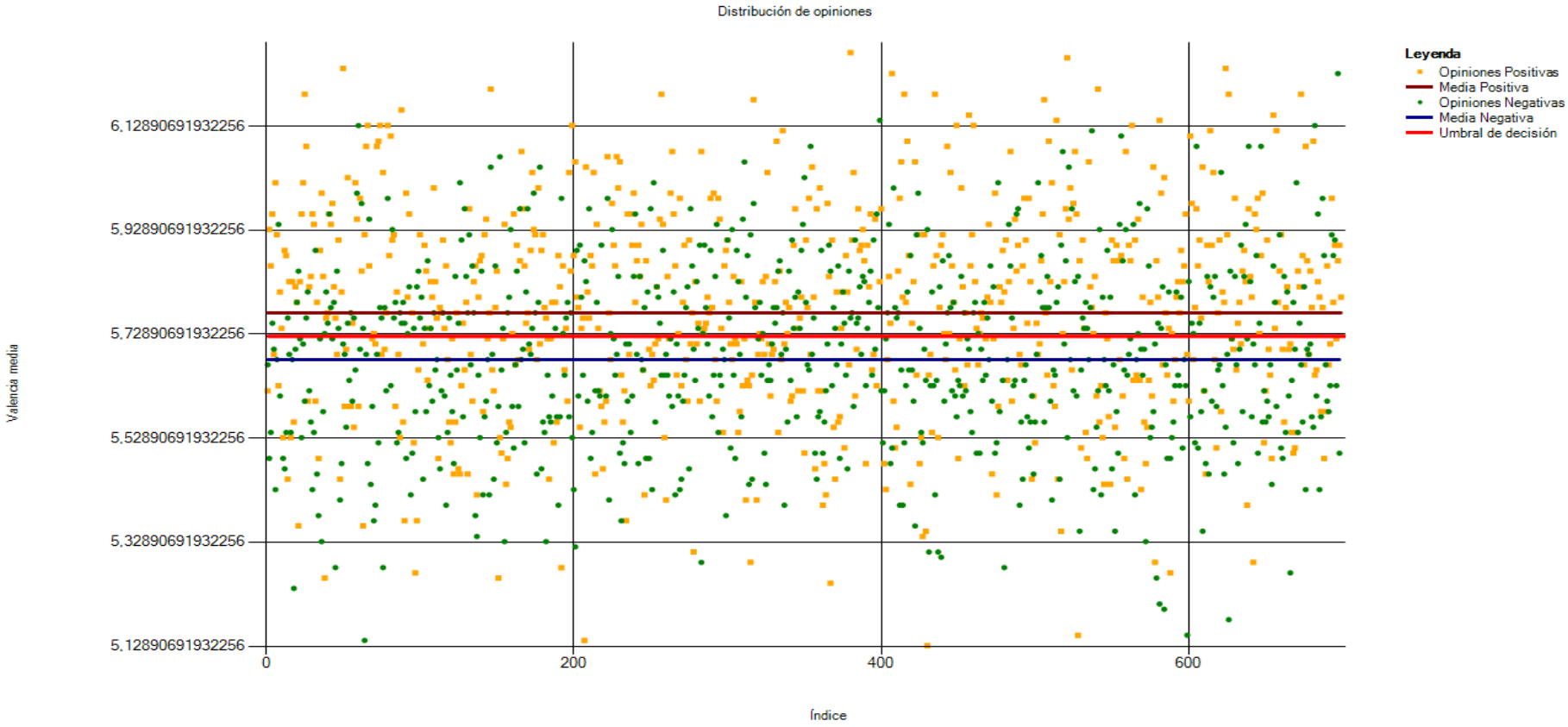

Figura 48 Gráfica de evaluación del clasificador con el conjunto de entrenamiento en formato blogging.

La leyenda de la gráfica en este caso es la misma que en el anterior. Para ver un video demostrativo pinchar en el link https://www.youtube.com/watch?v=M5dGkW-3FWw&list=UUYNmcO-KAHchzJxI7hup0lw .

4.4. Evaluación de opiniones manualmente

También se desarrolló una interfaz gráfica para realizar una clasificación no discreta, en el sentido "positiva" o "negativa", sino numérica en una escala entre 1 y 9. No obstante pese al valor numérico obtenido como se verá más adelante, fácilmente se puede convertir a una clase discreta del tipo "positiva" o "negativa", simplemente aplicando un umbral de decisión.

Cabe señalar que en este tipo de evaluación manual, no utilizamos las medidas de evaluación vistas anteriormente porque tiene un carácter empírico y unitario pues solamente se evalúa una opinión cada vez.

4.4.1. Resultados

En los siguientes ejemplos positivos y negativos introducidos manualmente se evalúa la efectividad del clasificador a vista.

Unos cuantos ejemplos positivos escritos por mi:

Figura 49 Resultado de la evaluación manual de la opinión positiva *"I love good teamwork with my colleagues"*.

Figura 50 Resultado la de evaluación manual de la opinión positiva de *"my mother's food is very delicious"*.

A continuación varias opiniones negativas introducidas también manualmente:

Figura 51 Resultado de la evaluación manual de la opinión negativa de "*I hate waiting at the bus stop*".

Figura 52 Resultado de la evaluación manual de la opinión negativa de "*corrupt politicians are worse than scum off the streets*".

Las opiniones positivas introducidas manualmente por mi "*I love good teamwork with my colleagues*" y "*my mother's food is very delicious*" recibieron las clasificaciones 7,38 y 7,45 respectivamente. Estos valores de clasificación son muy buenos porque estas frases están en formato microblogging y su valor de clasificación obtenido excede el umbral de decisión 5,795 ampliamente. Las opiniones negativas "*I hate waiting at the bus stop*" y "*corrupt politicians are worse than scum off the streets*" recibieron las clasificaciones 3,94 y 2,56 respectivamente. También son muy buenos valores de clasificación porque la polaridad emocional de dichas opiniones es negativa y como se ve los valores de clasificación están por debajo de 5,795 el umbral de decisión para el formato microblogging. Para ver un video demostrativo pinchar en el link https://www.youtube.com/watch?v=mfrFPPcqG3k .

4.5. Evaluación de opiniones generadas aleatoriamente

Para probar las opiniones generadas aleatoriamente se siguen los pasos siguientes:

1. Se selecciona el archivo de entrenamiento del modelo de la cadena de Markov.
2. Se tokeniza el texto contenido en el archivo.
3. Se genera el texto aleatorio.
4. Se clasifica.

A continuación, se mostrarán varios ejemplos positivos generado a partir de una crítica de cine positiva y negativos generados a partir de una crítica negativa, por lo cual, al menos en principio el texto generado aleatoriamente debe heredar la polaridad emocional del texto de entrenamiento de la cadena de Markov. También truncaremos el número de caracteres como máximo a 144 y a 1000 puesto que más que ello es innecesario.

Las dos siguientes imágenes se corresponden con textos aleatorios generados a partir de la misma opinión positiva de la película "*Carlito´s way*":

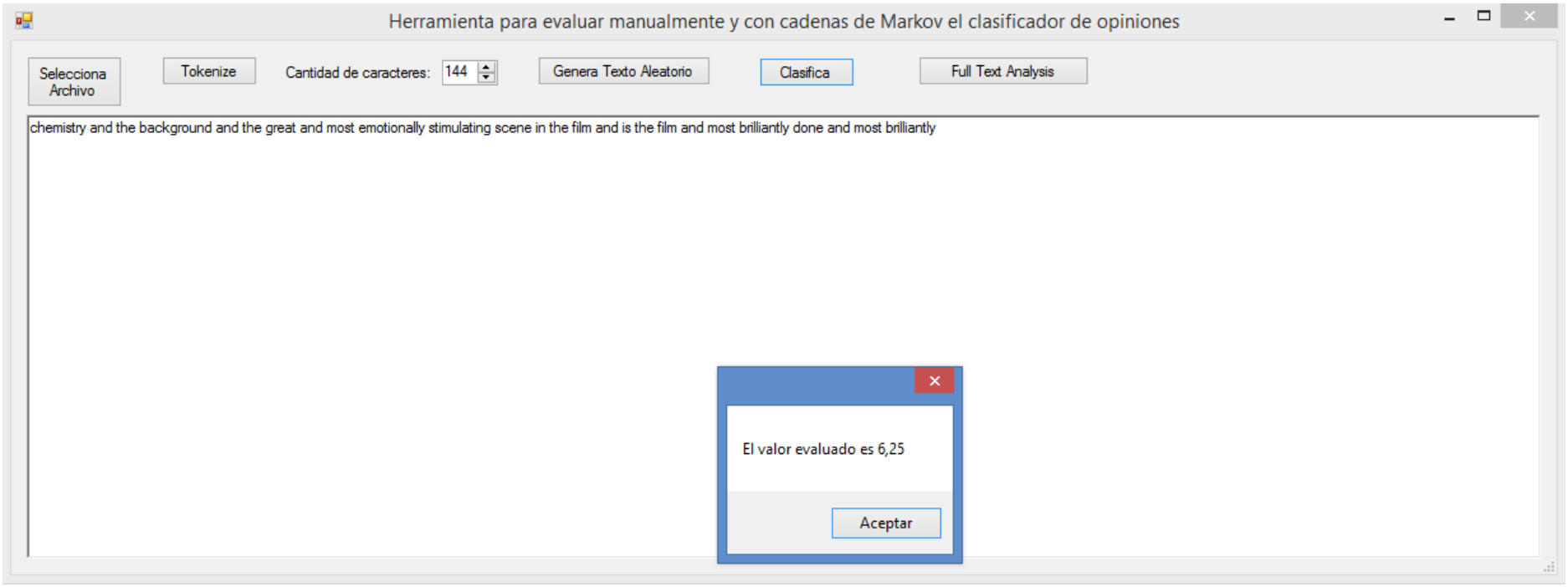

Figura 53 Evaluación de opinión generada aleatoriamente "*chemistry and the background and the great and most emotionally stimulating scene in the film and is the film and most brilliantly done and most brilliantly*" a partir de opinión positiva.

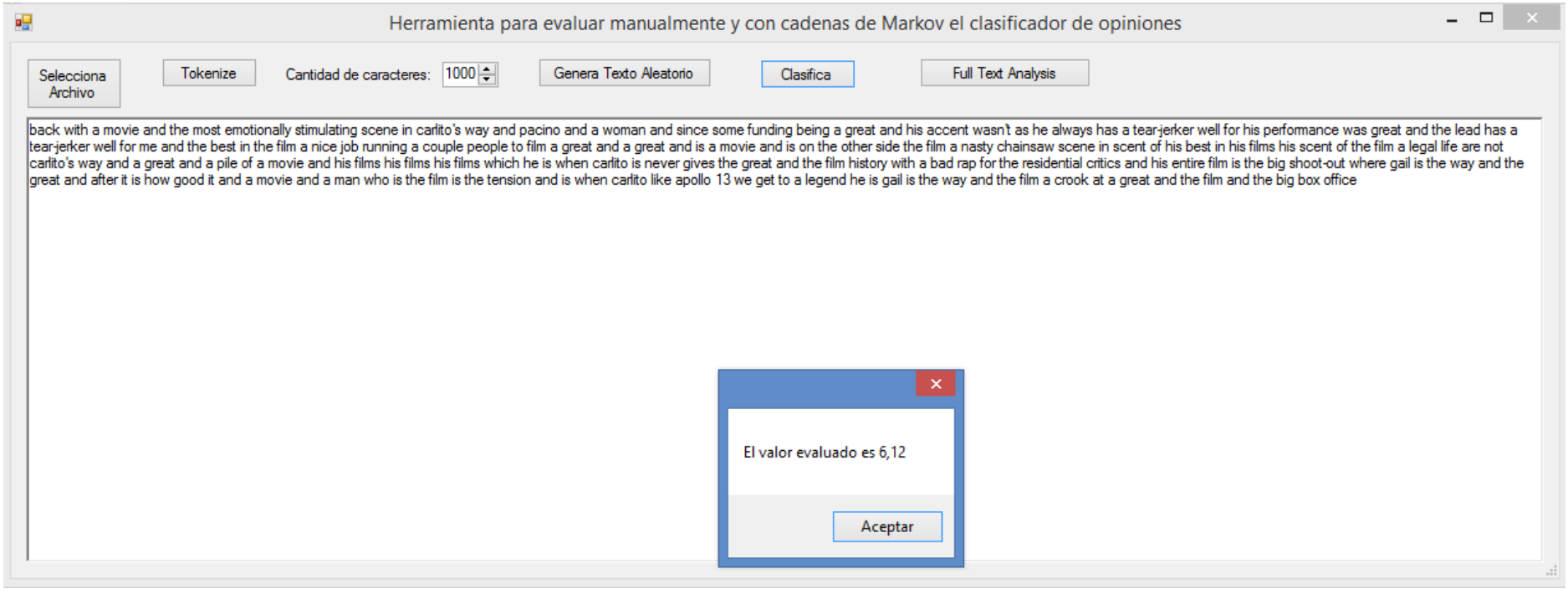

Figura 54 Evaluación de la opinión generada aleatoriamente "*back with a movie and the most emotionally stimulating scene in carlito's way and pacino and a woman and since some funding being a great and his accent wasn't as he always has a tear-jerker well for his performance was great and the lead has a tear-jerker well for me and the best in the film a nice job running a couple people to film a great and a great and is a movie and is on the other side the fil a nasty chainsaw scene in scent of his best in his hilms his scent of the film a legal life are not carlito´s way and a great and a pile of movie and his films his films his films which he is when carlito is never gives the great and the film history with a bad rap for the residential critics and his entire films is the big shoot-out where gail is the way and the fewat and after it is how good it and a movie and a man who is the film is the tension and is when carlito like apollo 13 we get to a legend he is gail is the way and the film a crook at a great and the film and the big box office*" a partir de opinión positiva.

Las dos siguientes evaluaciones, se realizan sobre la crítica negativa de la película "*The especialist*":

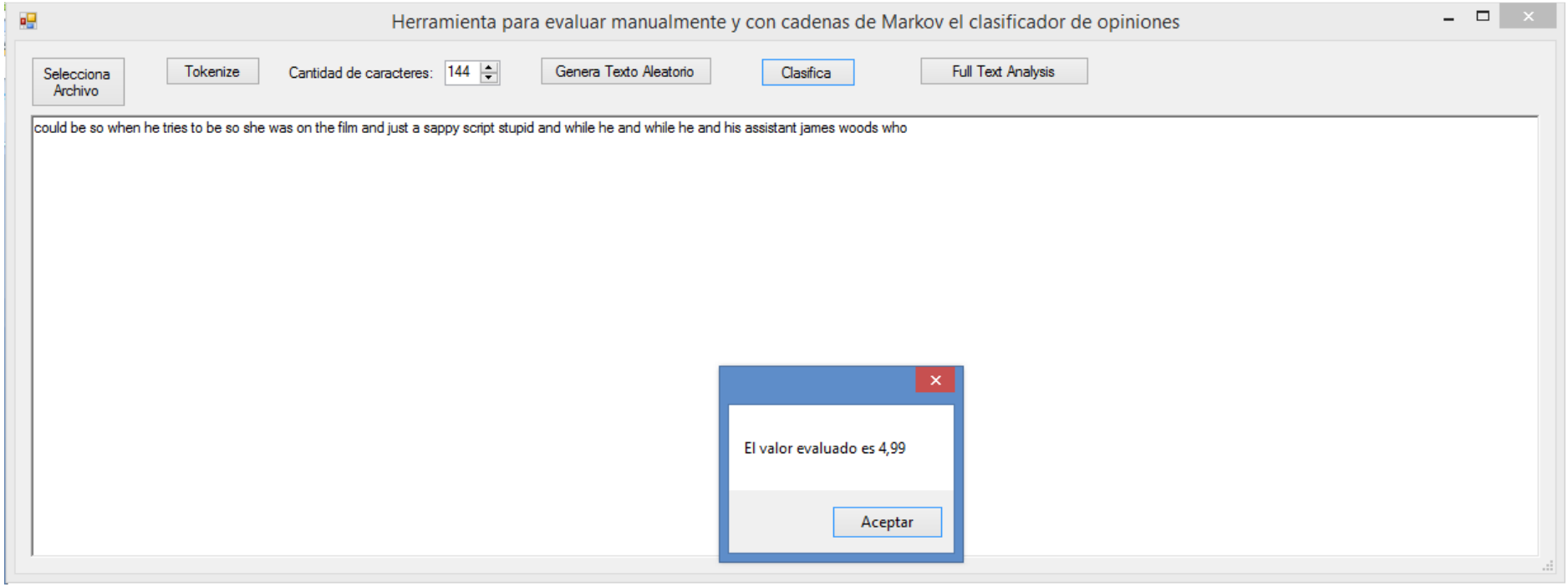

Figura 55 Evaluación de la opinión generada aleatoriamente *" could be so when he tries to be so she was on the film and just a sappy script stupid and while he and while he and his assistant james woods who "* a partir de opinión negativa.

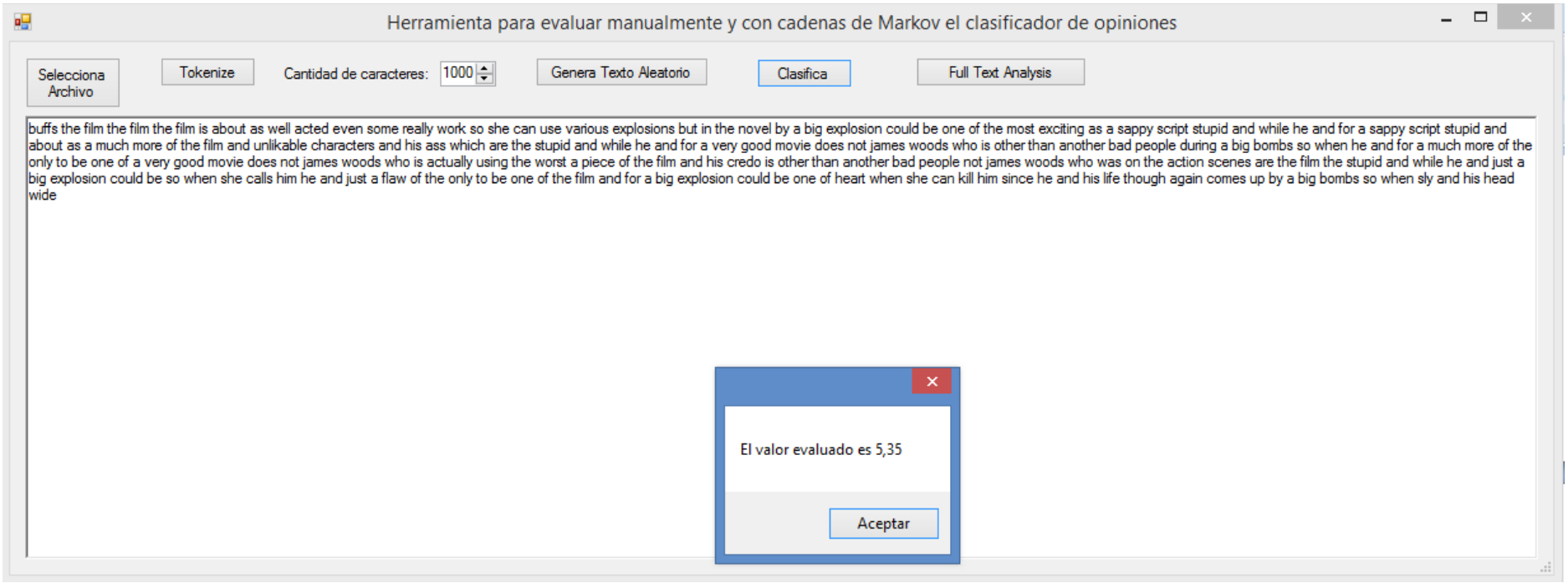

Figura 56 Evaluación de la opinión generada aleatoriamente *" buffs the film the film the film is about as well acted even some really work so she can use various explosions but in the novel by a big explosion could be one of the most exciting as a sappy script stupid and while he and for a sappy script stupid and about as a much more of the film and unlikable characters and his ass which are the stupid and while he and for a very good movie does not james woods who is other than another bad people during a big bombs so when he and for a much more of the only to be one of a very good movie does not james woods who is actually using the worst a piece of the film and his credo is other than another bad people not james woods who was on the action scenes are the film the stupid and while he and just a big explosion could be so when she calls him he and just a flaw of the only to be one of the film and for a big explosion could be one of heart when she can kill him since he and his life though again comes up by a big bombs so when sly and his head wide "* a partir de opinión negativa.

Los resultados de evaluación de textos generados aleatoriamente a partir de opiniones positivas o negativas son:

- La valencia media de la opinión generada aleatoriamente "*chemistry and the background and the great and most emotionally stimulating scene in the film and is the film and most brilliantly done and most brilliantly*", es 6,25 siendo este un valor positivo, ya que es mayor que 5,795 el umbral de decisión para textos en formato microblogging. Esto en cierta medida se debe a que el texto de entrenamiento para construir la cadena de Markov es a su vez positivo, y por ende se nota cierta tendencia a generar textos aleatorios también con dicha polaridad emocional.
- La valencia media de la opinión generada aleatoriamente "*back with a movie and the most emotionally stimulating scene in carlito's way and pacino and a woman and since some funding being a great and his accent wasn't as he always has a tear-jerker well for his performance was great and the lead has a tear-jerker well for me and the best in the film a nice job running a couple people to film a great and a great and is a movie and is on the other side the fil a nasty chainsaw scene in scent of his best in his hilms his scent of the film a legal life are not carlito´s way and a great and a pile of movie and his films his films his films which he is when carlito is never gives the great and the film history with a bad rap for the residential critics and his entire films is the big shoot-out where gail is the way and the fewat and after it is how good it and a movie and a man who is the film is the tension and is when carlito like apollo 13 we get to a legend he is gail is the way and the film a crook at a great and the film and the big box office*", es 6,12, lo cual es positivo también porque es mayor que el umbral de decisión de formato blogging, el cual es 5,725. En este caso se aprecia también la influencia positiva del texto de entrenamiento de la cadena de Markov.
- La valencia media de la opinión generada aleatoriamente "*could be so when he tries to be so she was on the film and just a sappy script stupid and while he and while he and his assistant james woods who "*, es 4,99 siendo negativa porque es menor que el umbral de decisión para el formato microblogging 5,795. Como se ve está influenciado por el carácter negativo del texto de entrenamiento (crítica a la película "The especialist").
- La valencia media de la opinión generada aleatoriamente *" buffs the film the film the film is about as well acted even some really work so she can use various explosions but in the novel by a big explosion could be one of the most exciting as a sappy script stupid and while he and for a sappy script stupid and about as a much more of the film and unlikable characters and his ass which are the stupid and while he and for a very good movie does not james woods who is other than another bad people during a big bombs so when he and for a much more of the only to be one of a very good movie does not james woods who is actually using the worst a piece of the film and his credo is other than another bad people not james woods who was on the action scenes are the film the stupid and while he and just a big explosion could be so when she calls him he and just a flaw of the only to be one of the film and for a big explosion could be one of heart when she can kill him since he and his life though again comes up by a big bombs so when sly and his head wide "*, es 5,35 siendo negativo ya que es menor que

5,725 el umbral de decisión para el formato blogging. Como se puede ver, también está influenciado por el hecho de que el texto de entrenamiento es negativo.

Para ver un video demostrativo pinchar en el link:
https://www.youtube.com/watch?v=pm0luyqZW5c&list=UUYNmcO-KAHchzJxI7hup0lw .

4.6. Análisis estadísitico

4.6.1. Formato microblogging

El hecho de que con el conjunto de entrenamiento en este formato de unas 7086 opiniones humanas, extraídas de redes sociales, el clasificador de opiniones obtenga un 84% de acierto global es un resultado buenísimo porque según estuve leyendo en un artículo [64] el análisis automatizado de sentimientos nunca va a ser más preciso que el análisis humano, ya que no se da cuenta de las sutilezas del sarcasmo o el lenguaje corporal o las jergas propias de una comunidad parlante. Sin embargo, de acuerdo a la experiencia de Biz360 (http://www.biz360.biz/) con *Mechanical Turk* [37], los seres humanos sólo están de acuerdo en el 79% de las veces. Eso significa que incluso cuando la exactitud en bruto de análisis automatizado de sentimientos es muy por debajo de lo perfecto, estadísticamente, puede ser pensado como más precisa en comparación con el análisis humano. En otras palabras, el análisis automatizado puede ser casi tan bueno como el análisis humano (o, "tan bueno como es posible"). Los humanos somos polémicos por naturaleza, siendo muy frecuente que exista una gran divergencia a la vez que convergencia en cuanto a temas se trata. Por ello el análisis automatizado del clasificador intenta determinar la polaridad emocional consensuada para cada opinión de manera genérica.

Una razón por la cual se alcanza el 84% de acierto es porque los conjuntos de opiniones positivas y negativas en formato *microblogging* son heterogéneos, y ello se demuestra en que la valencia media positiva es 6,78 mientras que la negativa 4,81. A mayor distancia entre ambas medias y menor desviación estándar de la muestra mejores clasificaciones obtendremos. De ahí el 84% de acierto total.

No solo se trata de que el 84% es un índice de acierto muy alto, sino de que está balanceado el acierto de ambas clases de predicción, siendo el porcentaje de acierto positivo 87% y el negativo 79%. Esto se debe a que el umbral de decisión para el formato de *microblogging* se eligió como el valor intermedio entre la valencia media global positiva y negativa.

La razón por la cual el porcentaje de acierto positivo es mayor que el negativo, es porque la desviación estándar de valencia media positiva es 0,88 mientras que la desviación estándar de la valencia media negativa es 1,36. Esto significa que la valencia media de las opiniones positivas se comporta de manera más uniforme que en las negativas, en otras palabras, que varía menos. Esto indica que las valencias medias positivas se acumulan más alrededor de la valencia media positiva global, es decir, de todo el conjunto de opiniones positivas, que en el caso de las negativas respecto a su media global negativa.

A continuación dos imágenes que demuestran esto:

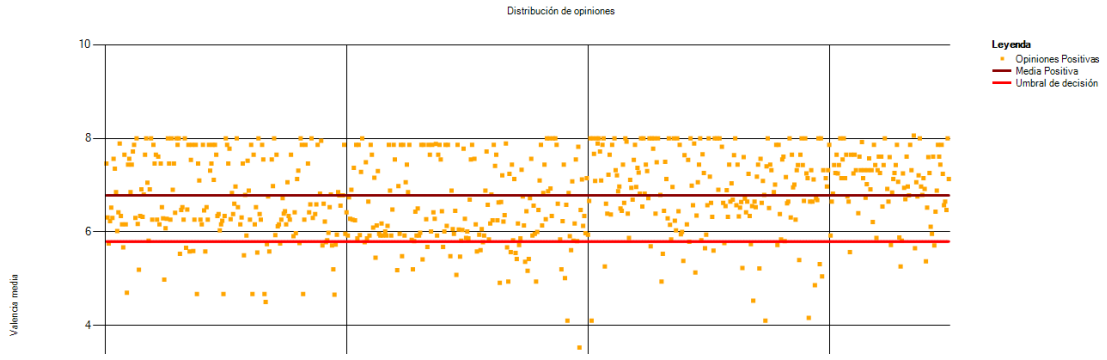

Figura 57 Distribución de opiniones positivas alrededor de la valencia media positiva global.

Si os fijáis veréis que son muy pocos las opiniones positivas, cuya valencia media está por debajo de la línea roja (el umbral de decisión). Otra observación muy importante es que casi todas las opiniones positivas están muy cerca de la valencia media positiva y ello es debido a lo dicho anteriormente, pues su desviación estándar es solo de 0,88.

Sin embargo miren la siguiente imagen:

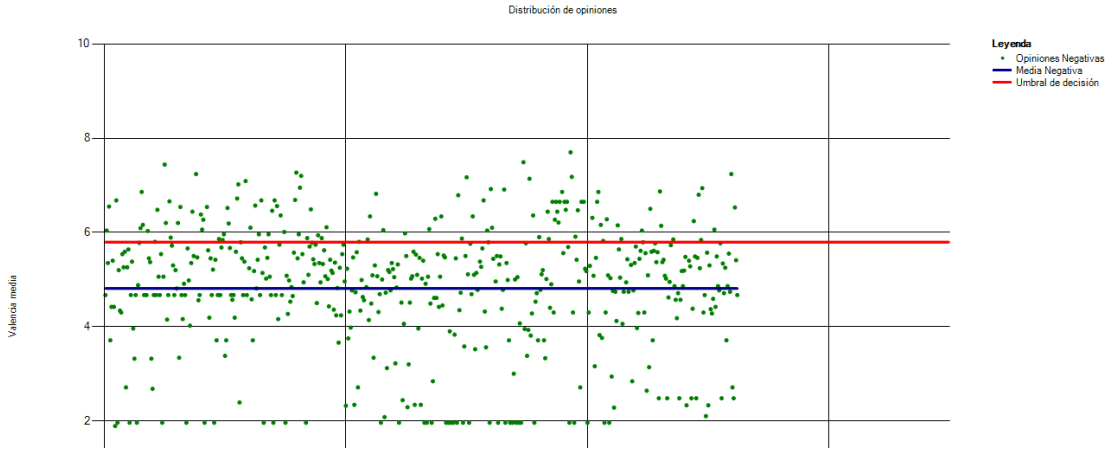

Figura 58 Distribución de opiniones negativas alrededor de la valencia media negativa global.

Nótese que el rango de la distribución de las valencias medias alrededor de la valencia media negativa global (línea azul) es bastante amplio en comparación con la gráfica de las opiniones positivas anterior. También, si os fijáis notaréis que hay más puntos verdes (opiniones negativas) por encima de la línea roja (umbral de decisión) que puntos anaranjados por debajo de la misma en la Figura 51.

Todo ello explica por qué el porcentaje de acierto positivo es ligeramente superior al negativo en un 8%.

Existe un 15% de textos de opiniones sin clasificar por mi algoritmo, porque ninguna de las palabras que contiene pertenece a la lista de palabras afectivas. Cabe señalar, que algunos textos están repetidos, los cuales solamente se clasificarían una vez el primero de ellos.

4.6.2. Formato blogging

El resultado del porcentaje de acierto para este formato de texto, es mucho menor que el 84% anterior. Estamos hablando de un 59% de acierto global con un 60% de acierto positivo y un 59% de acierto negativo. No obstante, es un buen resultado teniendo en cuenta, que la distribución de valencias medias para este formato del conjunto de opiniones positivas y negativas son mucho menos diferenciables (homogéneas), que en el caso de microblogging, como se verá a continuación:

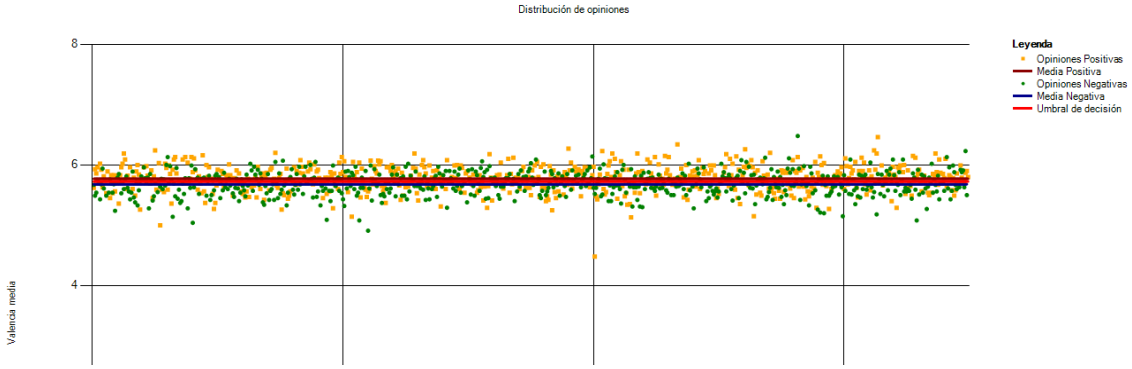

Figura 59 Distribución más homogénea de valencias medias entre los conjuntos de opiniones positivas y negativas.

Como se puede apreciar, ambas distribuciones de puntos se solapan y las medias casi coinciden. Para ver mejor la diferencia entre ambas distribuciones es necesario hacer un zoom a una escala de ampliación mayor.

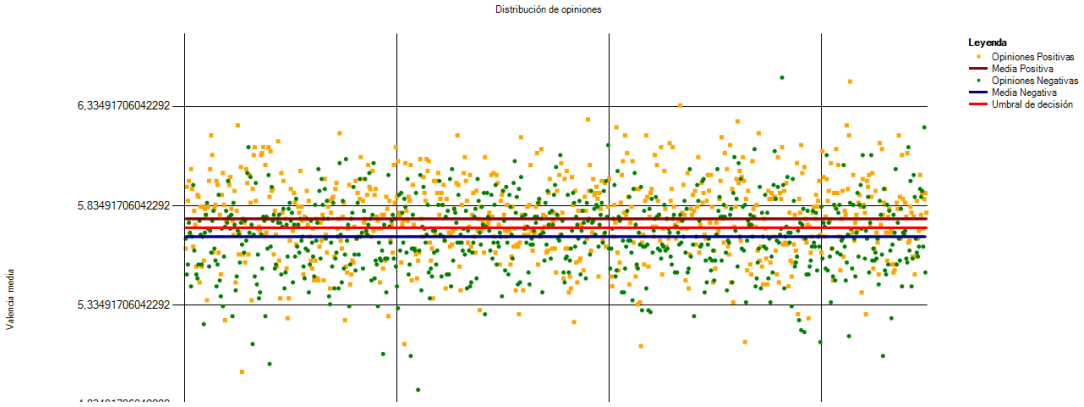

Figura 60 Primer zoom a la distribución de opiniones en formato blogging.

Ahora se distinguen un poco mejor pero aún así es necesario realizar otro zoom.

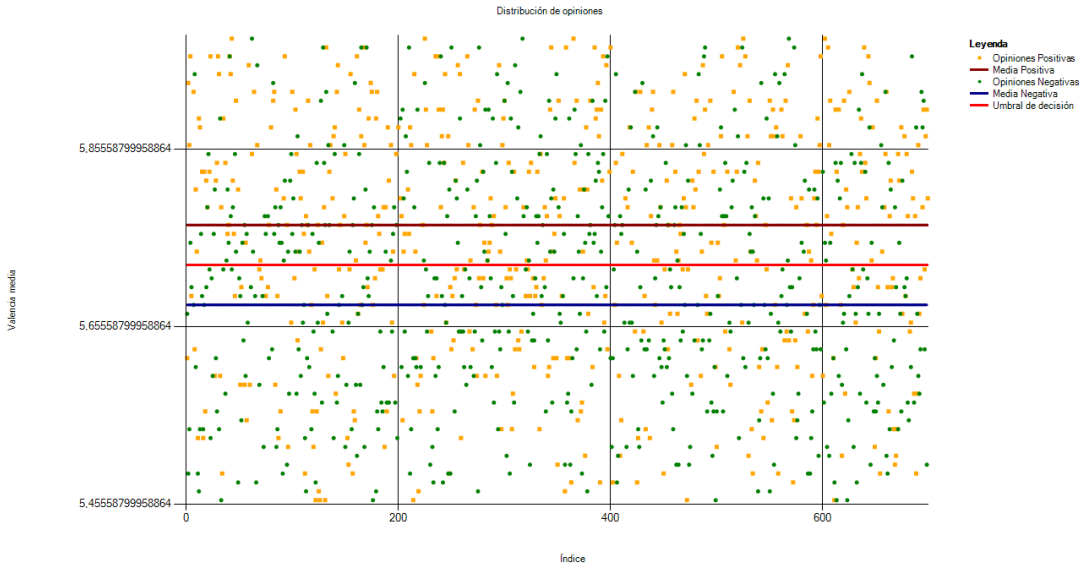

Figura 61 Zoom definitivo a la distribución de opiniones en formato blogging.

El hecho de que la valencia media positiva es 5,77 y la negativa 5,68 nos confirma lo que estamos viendo con nuestros propios ojos, que ambos conjuntos de opiniones son muy heterogéneos estadísticamente. De ahí que el porcentaje de acierto baje del 84% al 59%, es decir, descienda en un 25%.

Pese a ello un 59% de acierto no es un valor despreciable. Significa en que en más de la mitad de las opiniones el clasificador coincide con el conjunto de entrenamiento. Otro dato valioso es que están balanceados los porcentajes de acierto positivo y negativo, siendo 60% y 59% respectivamente. Ello se debe al umbral de decisión 5,725 aplicado de la misma manera que en el formato microblogging, como el valor intermedio entre las valencias medias positiva y negativa. Pero también se debe a que las desviaciones estándares de ambos conjuntos de opiniones positivas y negativas son muy pequeñas, siendo 0,21 y 0,19 respectivamente.

4. Planificación y presupuesto

La planificación del proyecto se divide en varias fases que conforman el ciclo de vida de la gestión del proyecto. Para cumplir en un tiempo conveniente los objetivos trazados con el proyecto, fue necesario planificar todas las actividades desarrolladas en cada fase.

A su vez, cada fase del ciclo de vida tiene coste asociado y recursos los cuales conforman el presupuesto final del proyecto.

5.1. Planificación

A continuación se muestran las fases determinadas que componen todo el proceso:

- Definición y análisis del problema: Estudio inicial del problema planteado, necesidades a solventar, estudio de plataformas y herramientas a utilizar y planteamiento de objetivos y límites temporales.
- Análisis del sistema: Definición de casos de uso y requisitos del software.
- Diseño del sistema: Definición de la arquitectura del sistema y componentes que lo forman.
- Implementación del sistema: Fase de codificación y desarrollo del sistema, adecuándolo a lo definido en las fases anteriores.
- Validación del sistema: Fase de realización de diversas pruebas que verifiquen la integridad, usabilidad y eficiencia del sistema con el objetivo de eliminar posibles errores.
- Documentación: Redacción de la memoria del proyecto.
- Presentación: Diseño de la presentación del proyecto y preparación de la lectura del mismo.

Figura 62 Esquema del ciclo de vida incremental del proyecto.

El paradigma de desarrollo en cascada [65] sigue siendo uno de los más utilizados a día de hoy. Consiste en las siguientes características:

- Cada fase empieza cuando se ha terminado la anterior.
- Para pasar a la fase posterior es necesario haber logrado los objetivos de la previa.
- Es útil como control de fechas de entregas.
- Al final de cada fase el personal técnico y los usuarios tienen la oportunidad de revisar el progreso del proyecto.

En cambio, para nuestro proyecto se ha elegido el paradigma de desarrollo incremental [65], de refinamiento sucesivo o mejora iterativa, ya que aunque posee las mismas fases del desarrollo en cascada, no tiene la restricción de linealidad, pues permite el solapamiento de fases, lo cual es muy útil cuando es necesario refinar un elemento de una fase anterior una vez comenzada la posterior. De este modo se garantiza una retroalimentación iterativa dentro del ciclo de vida del proyecto.

- **Diagrama de Gantt**

Partiendo de la base de que se ha elegido el modelo de ciclo de vida incremental para este proyecto, se procede a mostrar las planificaciones inicial y final.

5.1.1. Planificación inicial

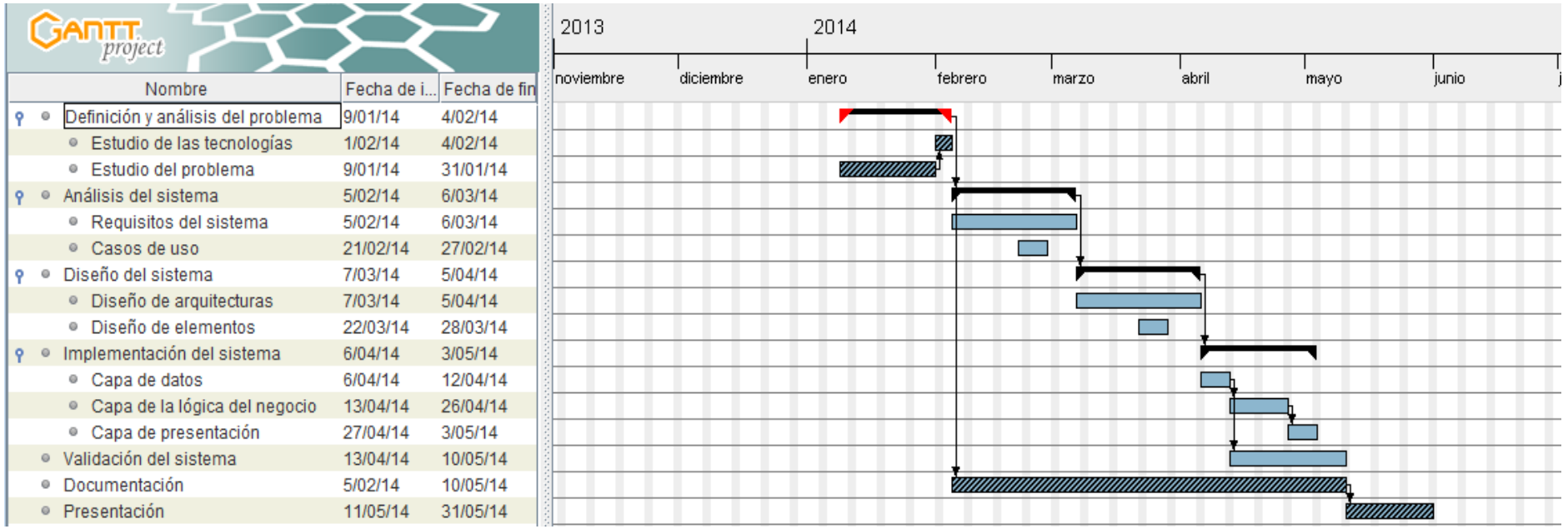

Figura 63 Diagrama de gantt de la planificación inicial del proyecto.

Como se puede apreciar, en esta planificación que comenzó a principios de Enero del 2014 la estimación de la duración total del proyecto es de casi 5 meses, dedicando entre 4 y 5 horas diarias, puesto que cuando comencé el proyecto trabajaba a media jornada laboral. Se pueden observar cada una de las fases del ciclo de vida del proyecto desglosadas en subtareas. Cada fase a su vez, es una tarea y cada tarea tiene una duración medida en días. Los fines de semanas están incluidos en los intervalos de los días de cada tarea. También se ven las dependencias entre las tareas, dado que unas deben terminar antes que otras, mientras que otras tareas pueden solaparse en el tiempo. Con esta planificación inicial, esperaba poder presentar y defender el proyecto en el mes de junio. Las tareas sombreadas representan la ruta crítica del proyecto.

5.1.2. Planificación final

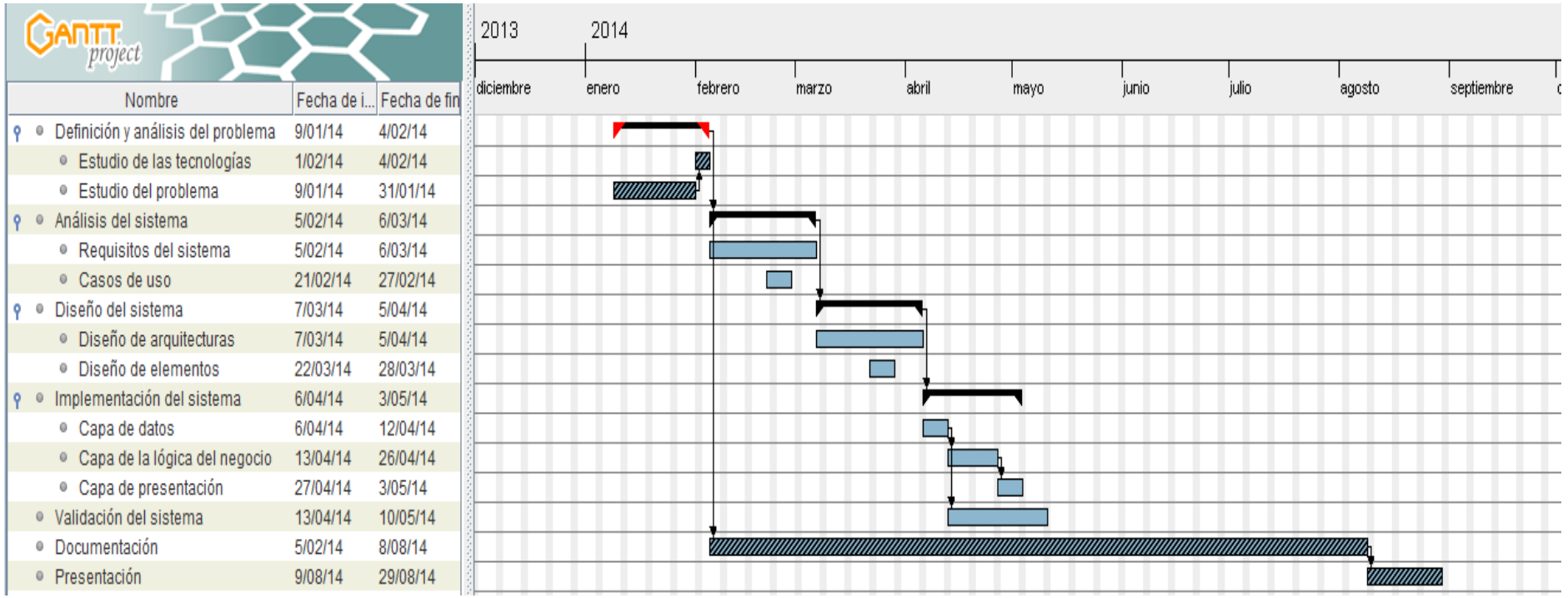

Figura 64 Diagrama de gantt de la planificación final del proyecto.

Es de destacar que la única diferencia entre la planificación inicial y la final es la duración de la tarea de documentación de la memoria del proyecto, la cual se alargó 3 meses por encima de lo planificado inicialmente. Ello se debe a que entre el mes de abril y mayo del 2014, comencé a trabajar a jornada intensiva de 8 horas, lo cual limitó las 4 horas diarias que podía dedicar antes a la mitad, o sea, 2 horas diarias como promedio aproximadamente. Esta es la razón por la cual defiendo el proyecto en el mes de Septiembre del 2014.

5.2. Presupuesto

En esta sección se van a detallar los costes derivados de la realización de las tareas detalladas anteriormente por parte del personal, junto con los surgidos por material necesario para la realización de las mismas.

5.2.1. Coste del personal encargado del proyecto

Los encargados del proyecto son el tutor y el alumno, donde se puede diferenciar los roles de cliente para el primero y el de diseñador, analista, programador y jefe de proyecto, dependiendo de la fase del ciclo de vida del mismo, para el segundo.

En el apartado de planificación final, vimos como la ruta crítica del proyecto comenzaba el 9 de Enero del 2014 y terminaba el 29 de Agosto del 2014, para un monto total de 232 días. Asumiendo una media diaria de 2 horas, debido a lo explicado en el apartado de planificación final, se obtiene un total de 464 horas dedicadas al proyecto por parte del alumno. Nótese que la mayor parte de las horas se dedicaron precisamente los fines de semana, porque entre lunes y viernes, era casi imposible dadas las circunstancias de mi trabajo y también el tiempo de ir y venir todos los días.

Lo anterior queda resumido en la siguiente tabla de costes totales:

Rol	Horas	Coste por hora	Coste total
Jefe de proyecto	46	40 €	1.840 €
Arquitecto	93	25 €	2.325 €
Analista	116	20 €	2.320 €
Desarrollador	209	13 €	2.717 €
Total			9.202 €

Tabla 21 Tabla de costes totales de personal.

En total, el gasto obtenido derivado de costes de personal al completar el proyecto es de 9.202 euros. Cabe señalar que las tarifas de coste por hora las estimé basándome en mi propia experiencia laboral personal y asumiendo la modalidad de *freelance* o autónomo.

5.2.2. Coste de material utilizado en el proyecto

A continuación, se muestra el resumen de costes de materiales del proyecto:

Material	Precio	Período de amortización	Duración del proyecto	Coste total
Equipo Intel(R) Core(TM) I3 CPU a 2.40 Gh cada procesador	530 €	36 meses	8 meses	117,77 €
Impresora HP C3180 Photosmart	160 €	36 meses	8 meses	35,56 €
Licencia de Microsoft Office 2007	129,99 €	36 meses	8 meses	28,89 €
Licencia de	0 €	36 meses	8 meses	0 €

Visual Studio 2010				
Licencia de Microsoft SQL Server 2005	0 €	36 meses	8 meses	0 €
Microsoft Windows 8.1	0 €	36 meses	8 meses	0 €
Total				182,22 €

Tabla 22 Tabla de costes totales de materiales en especial software y hardware.

Cabe señalar que las licencias del software de Visual Studio 2010 y Windows 8.1, son gratis gracias a la *Microsoft Academic Alliance*, la cual permite usar gratuitamente con fines académicos todo el software de Programa MSAA (*Microsoft Academic Alliance*) [66].

5.2.3. Coste total del proyecto

El coste total de realización del proyecto está dado por la siguiente tabla, donde se muestra el coste total derivado teniendo en cuenta tanto el personal como el material utilizado a lo largo del mismo.

Recurso	Coste
Personal	9.202 €
Material	182,22 €
Total	9384,22 €

Tabla 23 Tabla de costes totales del proyecto.

6. Conclusiones y trabajos futuros

Finalmente, termina el documento con una síntesis acerca del desarrollo de la aplicación y las conclusiones obtenidas a lo largo del mismo, añadiendo una propuesta de líneas futuras para mejora.

6.1. Conclusiones

Como se indicó en el capítulo de introducción, la motivación principal de este proyecto es detectar a tiempo la insatisfacción pública en el estado de opinión de la comunidad universitaria, en especial los estudiantes (a quiénes va dirigida la enseñanza), con el propósito de brindar una docencia de mayor calidad día a día.

El primer problema que encontré para la realización del clasificador de opiniones, fue que no encontré ninguna lista de palabras afectivas en lengua española similar a la que encontré [40] en lengua inglesa. Pese a ello, proseguí con el clasificador, porque existe una comunidad universitaria bilingüe (también hablan inglés) y aparte de esto siempre es posible traducir la opinión en español al inglés, para clasificarla en inglés y asignarle la misma clasificación, que a la correspondiente en lengua española. Lo ideal sería contar con una lista de palabras afectivas en lengua española.

El segundo problema al que me enfrenté, después de haber concebido el algoritmo de clasificación fue que no tenía una manera automática de evaluar la efectividad y precisión del algoritmo para clasificar la polaridad emocional de las opiniones humanas, excepto el modo

manual, que consistía en escribir manualmente los textos a clasificar en inglés y ver el valor de la clase devuelto. Para resolver este problema me ayudó mucho todo lo aprendido en la asignatura de Recuperación y Acceso a la Información, en la cual fue necesario evaluar la precisión y exhaustividad del motor de búsqueda que se realizó como parte de una práctica empleando la metodología de experimentación EIREX [67]. Es por ello que decidí experimentar con dos conjuntos de entrenamiento con textos en formato *microblogging* (textos cortos) y *blogging* (textos más largos). Como en estos conjuntos de entrenamiento, cada texto u opinión venían previamente clasificados en las dos clases "positiva" o "negativa", me vinieron como anillo al dedo para evaluar mi algoritmo. Una de las mayores ventajas, que proporciona este método de experimentación a gran escala (miles de instancias de entrenamiento), es que la evaluación del algoritmo no estaba comprometida con mi subjetividad, dado que he sido el desarrollador de este proyecto. Cabe destacar, que no fue fácil encontrar estos conjuntos de entrenamiento de manera gratuita en la red, con todos los requisitos que yo necesitaba. El primero que encontré fue el de formato *microblogging* y fue durante una clase de laboratorio de la asignatura de Aprendizaje Automático, ya que estuve investigando sobre algoritmos de aprendizaje automático que empleaban conjuntos de entrenamiento, los cuales se usan en herramientas como Weka [7] y así di con la competición patrocinada por la Universidad de Michigan, en la cual se proporcionaba el *training set* y el *test set*.

Cabe decir también, que al principio elegí el valor 5 como umbral de decisión, puesto que la valencia media de cada palabra afectiva oscila entre 1 y 9; por ende también la valencia media de una opinión oscilará en ese rango, y es por ello que al principio me pareció un buen candidato al umbral de decisión, por ser el valor intermedio entre 1 y 9. Entonces, comencé a ver que aunque mi porcentaje de acierto era alto en formato *microblogging*, un 76% de acierto, descubrí que en el caso de las opiniones negativas dicho porcentaje de acierto era inferior al 50%, mientras que en las positivas era casi el 100%. Observé en aquel momento también que rara vez la valencia media llegaba a ser 8, es decir que no llegaba a 9 ni tampoco encontré ninguna que llegara a 1 (siendo muy pocas las palabras afectivas que estuvieran en un extremo u otro), por lo cual llegué a la conclusión de que 5 como umbral de decisión no era la mejor opción y entonces decidí hacer un análisis estadístico de la distribución de la valencia media en la muestra positiva y la negativa, para afinar mi umbral de decisión según las medias de cada tipo de opinión. Fue entonces cuando mejoró muchísimo mi algoritmo, porque no solo aumentó el porcentaje de acierto global al 84%, sino que ahora sí clasificaba bien las opiniones negativas.

Otra dificultad encontrada con este proyecto fue la documentación del mismo, en especial en el apartado del estado de la cuestión, puesto que la información que realmente es relevante respecto al análisis de sentimientos y polaridad emocional, no es mucha porque no se han realizado tantos trabajos de minería de opiniones, según mi propia experiencia investigativa y comparándolo con otros temas de recuperación y acceso a la información o aprendizaje automático, los cuales están estrechamente relacionados con la clasificación de la polaridad emocional.

6.1.1. Sistema desarrollado

El sistema desarrollado se divide en varios componentes, cuyo propósito fundamental es permitir la evaluación del clasificador de opiniones, al cual le puse por nombre TBONTB, por ser el acrónimo de la famosa frase de Shakespeare “To be, or not to be...” esa es la cuestión, dado que mi clasificador trata precisamente de determinar la polaridad emocional de las opiniones, si son positivas o si no lo son, en cuyo caso son negativas.

El sistema posee dos interfaces gráficas fundamentales, una es para evaluar a gran escala el algoritmo de clasificación, mediante conjuntos de entrenamiento en formato *microblogging* o *blogging*, mostrando además una gráfica con la distribución de la valencia media de cada opinión, donde se puede realizar un análisis estadístico a simple vista de la muestra y ver qué opinión corresponde a cada valor de valencia media en la gráfica. La otra interfaz es para evaluar el clasificador pero a una escala mucho más pequeña, de manera manual, escribiendo o pegando el texto en inglés en la caja de texto y la otra manera es generando texto aleatorio mediante el uso de cadenas de Markov, las cuales se entrenan y construyen a partir de opiniones de texto en inglés en formato *blogging* (con varias oraciones de texto para entrenar la cadena de Markov) cuya polaridad es conocida. Dicho texto generado se clasifica también para observar las distintas tendencias de clasificación aleatoria como parte del proceso de evaluación.

6.1.2. Proceso de desarrollo

Es destacable que gracias a las reuniones realizadas con mi tutor, pude enfocar y encaminar bien primero que todo el propósito del proyecto (al principio no lo tenía muy claro), y luego el correcto análisis del sistema, extrayendo los requisitos adecuados y casos de uso según las necesidades del mismo, lo cual ha facilitado mucho el desarrollo del proceso de validación e implementación, logrando no solo alcanzar los objetivos iniciales planteados, sino también que sea fácil usar. El diseño del sistema no se dejó de la mano tampoco, resaltando la arquitectura por capas de acceso a datos, lógica del negocio, presentación (incluye gráficas de distribución de valencias medias y generador de texto aleatorio con cadenas de Markov) y por componentes del sistema. Prueba de ello son los resultados obtenidos en la evaluación del algoritmo de clasificación mediante la herramienta desarrollada.

6.1.3. Personales

En lo personal, estoy muy contento con mi tutor y mi proyecto, no solo por todo lo que he aprendido de la disciplina *Opinion Mining*, la cual me apasiona mucho, sino porque he obtenido resultados que han sobrepasado todas mis expectativas, especialmente el balanceo de acierto positivo y negativo, tanto en formato *microblogging* como *blogging*, así como también el porcentaje de acierto del 84% para formato *microblogging,* que es buenísimo teniendo en cuenta que la mayoría de las opiniones de los alumnos y profesores expresadas en foros de asignaturas tienden a ser más bien cortas que largas. El formato *microblogging* como sabéis es el más usado en las redes sociales y dada la diversidad del conjunto de entrenamiento en dicho formato empleado con 7086 opiniones diferentes el acierto obtenido en ese sentido ha sido estupendo.

Así mismo me siento muy agradecido, porque este proyecto me ha permitido integrar conocimientos de asignaturas muy diversas, consolidando todos mis conocimientos aprendidos a lo largo de la carrera y finalmente me regocija mucho saber que pronto me gradúo de ingeniero en informática.

6.2. Trabajos futuros

De cara al futuro para mejorar el porcentaje de acierto de las opiniones en formato *blogging*, el cual es 59%, le aplicaría un filtro semántico para solamente tener en cuenta en el cálculo de la valencia media del texto aquellas palabras afectivas, que realmente son relevantes para el tema del texto. Esto se puede lograr con Wordnet [68] en una versión de SQL server que contiene la siguiente tabla:

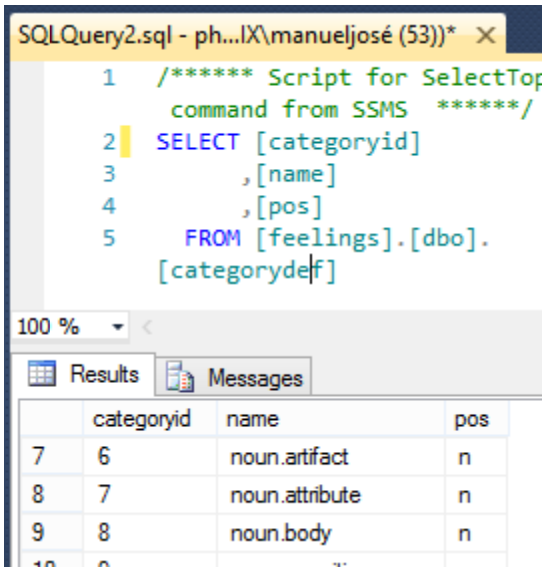

Figura 65 Tabla categorydef de Wordnet.

El campo categoryid de esta tabla aparece también en la tabla synset:

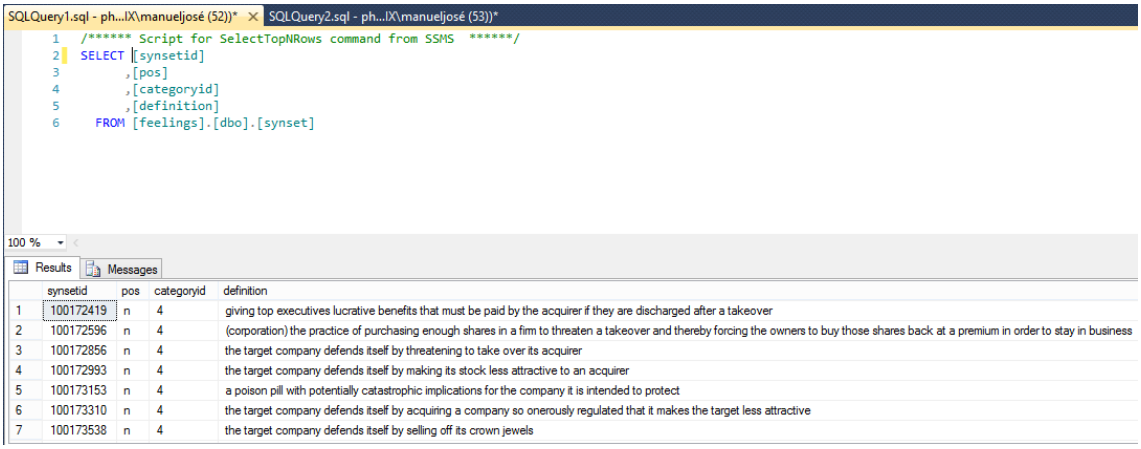

Figura 66 Tabla synset de Wordnet.

Eso significa que después de haber realizado el POS *tagging* [69], conociendo si la palabra afectiva en cuestión es un sustantivo, adjetivo, u otra parte gramatical de la oración se

identifica a qué synset pertenece, y por tanto se identifica también la categoría semántica por el campo catergoryid que es la llave de la tabla categorydef. Algunos ejemplos de categorías son animal, artefacto, atributo, cuerpo y muchos otros. Dado que estamos hablando de opiniones en formato *blogging* (textos con varias oraciones), es muy probable que en el conjunto de palabras afectivas reconocidas del texto se repitan varias categorías semánticas. La que más se repita es el tópico de la opinión y por tanto para aplicar el filtro semántico es necesario calcular la similitud semántica entre cada palabra afectiva (campo definition de la tabla synset) y el tópico, es decir, la categoría semántica (campo name de la tabla categorydef) que más se repite para eliminar de la ecuación que calcula la valencia media aquellas palabras afectivas que se encuentran a mayor distancia del tópico de la opinión, definiendo un umbral numérico de distancia con respecto a la similitud semántica.

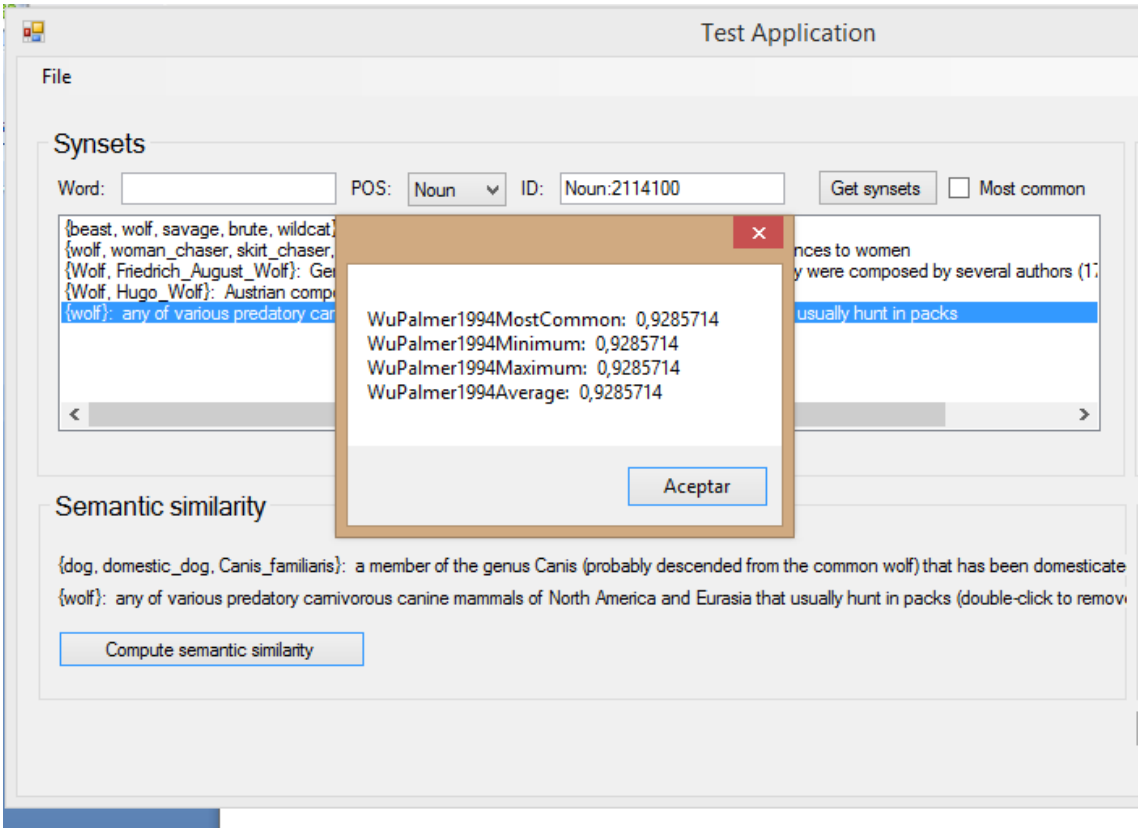

Figura 67 Similitud semántica entre synset de la palabra dog y wolf.

La Figura 65 es el *framework* Antelope [70], el cual es un entorno de procesamiento de lenguaje natural gratis y open source. Este framework lo he integrado en la solución de mi proyecto, al igual que Wordnet [71] en SQL server en la base de datos feelings. En la Figura 65 se muestra la similitud semántica entre dog (perro) y wolf (lobo) que es de un 92%, es decir, casi el 100%, lo cual es lógico. Sin embargo cuando se trata de la similitud entre hot dog y wolf esta similitud desciende mucho hasta 18% como se muestra a continuación:

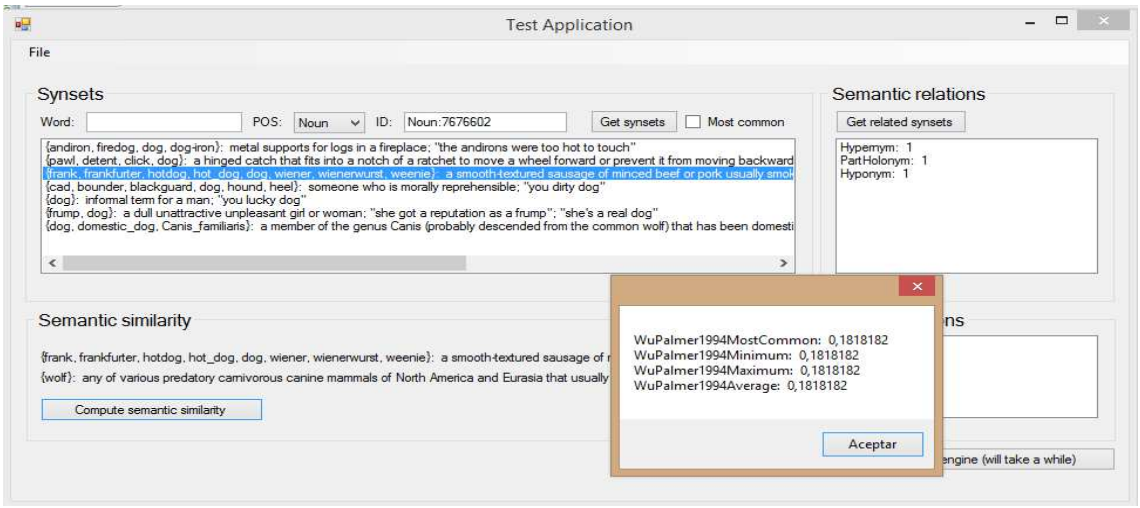

Figura 68 Similitud semántica entre synset de la palabra dog (cuando significa hot dog) y wolf.

El propósito de aplicar este filtro semántico a las opiniones en formato *blogging,* es evitar la homogeneidad en cuanto a la valencia media de las positivas y las negativas. De este modo se diferenciarían más las opiniones positivas de las negativas. Con respecto al valor de la media negativa sería más cercano a 1, mientras que el valor de la media positiva sería más cercano a 9. De este modo y asumiendo una desviación estándar menor a 0,5 pronostico un considerable aumento del porcentaje de acierto que actualmente es 59% para el formato *blogging*.

Otra posible mejora sería conociendo el género sexual, la edad, o el nivel educacional de la persona que ha escrito la opinión los valores de valencia media aplicados a las palabras afectivas, serían más precisos, ya que en la tabla de ANEW scores, donde se almacenan dichas palabras, además de la valencia media genérica, existe una valencia media genérica por género masculino y otra por género femenino. Igualmente ocurre con la edad, pues existen 3 valencias medias para los jóvenes, adultos y ancianos. Lo mismo para el nivel educacional. Aplicando estas valencias medias en vez de la genérica, se refinaría aún más la clasificación de la opinión.

Cabe destacar como otra línea de trabajo futuro, mediante el uso de la herramienta de aprendizaje automático Weka [7] la búsqueda de una fórmula similar a la de la Figura 9, que combine no solamente la valencia media de las palabras afectivas, sino también la dominancia media y el grado de excitación media, las cuales son dimensiones emocionales que se pueden emplear en la misma escala entre 1 y 9.

Finalmente, como última propuesta de trabajo futuro sería un valor añadido a la clasificación de la opinión, extraer los sentimientos reflejados explícitamente en el texto de la opinión, a partir de un listado de sentimientos de We Feel Fine [29], el cual se compararía con las palabras del texto de la opinión. De este modo, no solamente contaríamos con la clasificación positiva o negativa de la opinión, sino también con los sentimientos implicados explícitamente en la misma.

7.0. Bibliografía

[1] «Universidad Carlos III de Madrid,» [En línea]. Available: http://www.uc3m.es/Inicio.

[2] Ó. Ray, "http://unadocenade.com/," 16 mayo 2013. [Online]. Available: http://unadocenade.com/una-docena-de-claves-para-entender-la-importancia-del-fenomeno-big-data/. [Accessed febrero 2014].

[3] D. Szkolar, «http://infospace.ischool.syr.edu,» 24 enero 2013. [En línea]. Available: http://infospace.ischool.syr.edu/2013/01/24/data-mining-in-obamas-2012-victory/. [Último acceso: febrero 2014].

[4] "http://en.wikipedia.org," [Online]. Available: http://en.wikipedia.org/wiki/Sentiment_analysis. [Accessed febrero 2014].

[5] B. Liu, «http://www.cs.uic.edu/,» 21 abril 2008. [En línea]. Available: http://www.cs.uic.edu/~liub/FBS/opinion-mining-sentiment-analysis.pdf. [Último acceso: febrero 2014].

[6] P. J. L. Margaret M. Bradley, "http://citeseerx.ist.psu.edu," 1999. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.306.3881&rep=rep1&type=pdf. [Accessed febrero 2014].

[7] "Weka 3: Data Mining Software in Java," [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/. [Accessed febrero 2014].

[8] P. D. Turney, "http://acl.ldc.upenn.edu," julio 2002. [Online]. Available: http://acl.ldc.upenn.edu/P/P02/P02-1053.pdf. [Accessed febrero 2014].

[9] L. L. S. V. Bo Pang, "http://acl.ldc.upenn.edu/," julio 2002. [Online]. Available: http://acl.ldc.upenn.edu/acl2002/EMNLP/pdfs/EMNLP219.pdf. [Accessed febrero 2014].

[10] D. D. Lewis, "http://link.springer.com," 1998. [Online]. Available: http://link.springer.com/chapter/10.1007%2FBFb0026666#page-1. [Accessed febrero 2014].

[11] A. D. P. S. J. D. P. V. Berger, "http://dl.acm.org," 1996. [Online]. Available: http://dl.acm.org/citation.cfm?id=234289. [Accessed febrero 2014].

[12] T. Joachims, "http://dl.acm.org," 1998. [Online]. Available: http://dl.acm.org/citation.cfm?id=649721. [Accessed febrero 2014].

[13] E. W. J. Riloff, "http://www.cs.utah.edu/," 2003. [Online]. Available: http://www.cs.utah.edu/~riloff/pdfs/emnlp03.pdf. [Accessed febrero 2014].

[14] H. H. V. Yu, "Towards Answering Opinion Questions: Separating Facts," in *Proceedings of the 2003 Conference onEmpirical Methods in Natural Language Processing(EMNLP)*, Japan, Sapporo, 2003.

[15] M. L. B. Hu, "Mining and Summarizing Customer Reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, Seattle, WA, USA, 2004.

[16] Marc, "http://crr.ugent.be/," enero 2013. [Online]. Available: http://crr.ugent.be/archives/1003. [Accessed noviembre 2013].

[17] M. M. B. a. P. J. Lang, "http://citeseerx.ist.psu.edu/," 1999. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.306.3881&rep=rep1&type=pdf. [Accessed noviembre 2013].

[18] F. Å. Nielsen, "http://www2.imm.dtu.dk," 2011. [Online]. Available: http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/6028/pdf/imm6028.pdf. [Accessed febrero 2014].

[19] "http://www.saifmohammad.com/," [Online]. Available: http://www.saifmohammad.com/WebPages/ResearchInterests.html. [Accessed febrero 2014].

[20] "http://www.plosone.org," [Online]. Available: http://www.plosone.org/article/fetchSingleRepresentation.action?uri=info:doi/10.1371/journal.pone.0026752.s001. [Accessed abril 2014].

[21] "http://wndomains.fbk.eu," [Online]. Available: http://wndomains.fbk.eu/wnaffect.html. [Accessed enero 2014].

[22] "http://sentistrength.wlv.ac.uk/," [Online]. Available: http://sentistrength.wlv.ac.uk/. [Accessed abril 2014].

[23] "http://www.clips.ua.ac.be," [Online]. Available: http://www.clips.ua.ac.be/pages/pattern. [Accessed abril 2014].

[24] "https://code.google.com," [Online]. Available: https://code.google.com/p/sasa-tool/. [Accessed abril 2014].

[25] "https://senti.crowdflower.com/," [Online]. Available: https://senti.crowdflower.com/. [Accessed abril 2014].

[26] S. Grimes, "http://breakthroughanalysis.com," enero 2012. [Online]. Available: http://breakthroughanalysis.com/2012/01/08/what-are-the-most-powerful-open-source-sentiment-analysis-tools/?goback=.gde_2668649_member_204024865. [Accessed abril

2014].

[27] "Wee Feel Fine," [Online]. Available: http://www.wefeelfine.org/. [Accessed noviembre 2013].

[28] «http://www.wefeelfine.org,» [En línea]. Available: http://www.wefeelfine.org/data/files/feelings.txt. [Último acceso: noviembre 2013].

[29] "Wee Feel Fine," [Online]. Available: http://www.wefeelfine.org/api.html. [Accessed noviembre 2014].

[30] S. D. K. a. J. Harris, "http://www.wefeelfine.org/," [Online]. Available: http://www.wefeelfine.org/wefeelfine.pdf. [Accessed noviembre 2013].

[31] P. D. a. C. Danforth, «http://www.uvm.edu/,» 8 December 2011. [En línea]. Available: http://www.uvm.edu/~cdanfort/research/2011-hedonometer-arxiv.pdf. [Último acceso: November 2013].

[32] "Google Books," [Online]. Available: http://books.google.com/books?hl=en. [Accessed Abril 2014].

[33] "New York Times articles," [Online]. Available: http://www.nytimes.com/ref/membercenter/nytarchive.html.

[34] "Music Lyrics," [Online]. Available: http://www.lyrics.com/.

[35] "Twitter," [Online]. Available: https://twitter.com/.

[36] [Online]. Available: http://www.hedonometer.org/words.html.

[37] "Amazon´s Mechanical Turk," [Online]. Available: https://www.mturk.com/mturk/welcome.

[38] "hedonometer," [Online]. Available: http://www.hedonometer.org/index.html.

[39] P. D. a. C. Danforth, 17 July 2009. [Online]. Available: http://www.uvm.edu/~cdanfort/research/dodds-danforth-johs-2009.pdf. [Accessed November 2013].

[40] A. W. a. V. Kuperman, "http://crr.ugent.be/," 2013. [Online]. Available: http://crr.ugent.be/papers/Warriner_et_al_affective_ratings.pdf. [Accessed January 2014].

[41] E. B. f. S. S. a. Control, «ftp://ftp.estec.esa.nl,» February 1991. [En línea]. Available: ftp://ftp.estec.esa.nl/pub/wm/anonymous/wme/bssc/PSS050.pdf. [Último acceso: April 2014].

[42] [En línea]. Available: http://msdn.microsoft.com/es-es/library/8z6watww(v=vs.110).aspx.

[43] [En línea]. Available: http://buscon.rae.es/drae/?type=3&val=algoritmo&val_aux=&origen=REDRAE.

[44] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/bb399567(v=vs.110).aspx.

[45] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/hs600312(v=vs.110).aspx.

[46] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/bb762916(v=vs.110).aspx.

[47] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/bb386976(v=vs.110).aspx.

[48] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/bb397687.aspx.

[49] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/dd268536(v=vs.110).aspx.

[50] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/hk4ts42s(v=vs.90).aspx.

[51] "http://msdn.microsoft.com," [Online]. Available: http://msdn.microsoft.com/es-es/library/system.windows.forms.datavisualization.charting.chart(v=vs.110).aspx.

[52] "http://msdn.microsoft.com," [Online]. Available: http://msdn.microsoft.com/es-es/library/bb383977.aspx.

[53] "http://msdn.microsoft.com," [Online]. Available: http://msdn.microsoft.com/es-es/library/system.windows.forms.datavisualization.charting.chartarea(v=vs.110).aspx.

[54] "http://www.codeproject.com," [Online]. Available: http://www.codeproject.com/Articles/357817/MSChart-Extension-Zoom-and-Pan-Control.

[55] [Online]. Available: http://opensource.org/licenses/mit-license.php.

[56] "http://msdn.microsoft.com," [Online]. Available: http://msdn.microsoft.com/en-us/library/system.windows.forms.datavisualization.charting.chart.gettooltiptext(v=vs.110).aspx.

[57] "http://msdn.microsoft.com," [Online]. Available: http://msdn.microsoft.com/en-us/library/system.windows.forms.datavisualization.charting.datapoint(v=vs.110).aspx.

[58] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/ms187510.aspx.

[59] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/ms187048.aspx.

[60] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/ms254494(v=vs.110).aspx.

[61] «http://msdn.microsoft.com,» [En línea]. Available: http://msdn.microsoft.com/en-us/library/vstudio/k3bb4tfd(v=vs.100).aspx.

[62] [Online]. Available: http://opensource.org/licenses/MIT.

[63] «assembla,» [En línea]. Available: https://www.assembla.com.

[64] "tortoisesvn," [Online]. Available: http://tortoisesvn.net/downloads.html.

[65] «http://inclass.kaggle.com,» [En línea]. Available: http://inclass.kaggle.com/c/si650winter11.

[66] «http://www.cs.cornell.edu/,» [En línea]. Available: http://www.cs.cornell.edu/People/pabo/movie-review-data/.

[67] O. María, "http://mashable.com/," febrero 2014. [Online]. Available: http://mashable.com/2010/04/19/sentiment-analysis/.

[68] Z. L. F. P. R. G. M. R. Cataldi, «http://www.iidia.com.ar/,» [En línea]. Available: http://www.iidia.com.ar/rgm/comunicaciones/c-icie99-ingenieriasoftwareeducativo.pdf.

[69] «http://www.lab.inf.uc3m.es,» [En línea]. Available: http://www.lab.inf.uc3m.es/servicios/msdnaa.

[70] "http://uc3m.es," [Online]. Available: http://ir.kr.inf.uc3m.es/eirex/.

[71] T. Simpson, "http://opensource.ebswift.com," [Online]. Available: http://opensource.ebswift.com/WordNetSQLServer/.

[72] "http://www-nlp.stanford.edu," [Online]. Available: http://www-nlp.stanford.edu/links/statnlp.html#Taggers.

[73] "https://www.proxem.com," [Online]. Available: https://www.proxem.com/technologie/antelope/.

[74] "http://wordnet.princeton.edu/," [Online]. Available: http://wordnet.princeton.edu/.